

Synthese Library 480

Studies in Epistemology, Logic, Methodology,
and Philosophy of Science

Melvin Fitting
Richard L. Mendelsohn

First-Order Modal Logic

Second Edition

Synthese Library

Studies in Epistemology, Logic, Methodology,
and Philosophy of Science

Volume 480

Editor-in-Chief

Otávio Bueno, Department of Philosophy, University of Miami, Coral Gables, USA

Editorial Board Members

Berit Brogaard, University of Miami, Coral Gables, USA

Steven French, University of Leeds, Leeds, UK

Catarina Dutilh Novaes, VU Amsterdam, Amsterdam, The Netherlands

Darrell P. Rowbottom, Department of Philosophy, Lingnan University, Tuen Mun, Hong Kong

Emma Ruttkamp, Department of Philosophy, University of South Africa, Pretoria, South Africa

Kristie Miller, Department of Philosophy, Centre for Time, University of Sydney, Sydney, Australia

The aim of *Synthese Library* is to provide a forum for the best current work in the methodology and philosophy of science and in epistemology, all broadly understood. A wide variety of different approaches have traditionally been represented in the Library, and every effort is made to maintain this variety, not for its own sake, but because we believe that there are many fruitful and illuminating approaches to the philosophy of science and related disciplines.

Special attention is paid to methodological studies which illustrate the interplay of empirical and philosophical viewpoints and to contributions to the formal (logical, set-theoretical, mathematical, information-theoretical, decision-theoretical, etc.) methodology of empirical sciences. Likewise, the applications of logical methods to epistemology as well as philosophically and methodologically relevant studies in logic are strongly encouraged. The emphasis on logic will be tempered by interest in the psychological, historical, and sociological aspects of science. In addition to monographs *Synthese Library* publishes thematically unified anthologies and edited volumes with a well-defined topical focus inside the aim and scope of the book series. The contributions in the volumes are expected to be focused and structurally organized in accordance with the central theme(s), and should be tied together by an extensive editorial introduction or set of introductions if the volume is divided into parts. An extensive bibliography and index are mandatory.

Melvin Fitting • Richard L. Mendelsohn

First-Order Modal Logic

Second Edition

 Springer

Melvin Fitting
Graduate Center
City University of New York
New York, NY, USA

Richard L. Mendelsohn
Graduate Center
City University of New York
New York, NY, USA

ISSN 0166-6991

ISSN 2542-8292 (electronic)

Synthese Library

ISBN 978-3-031-40713-0

ISBN 978-3-031-40714-7 (eBook)

<https://doi.org/10.1007/978-3-031-40714-7>

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Switzerland AG 1998, 2023

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Paper in this product is recyclable.

To Benjamin, Josie, and Jillian, with love.

RLM

*To the memory of my parents, Helen and
Chris Fitting, and my thesis advisor
Raymond Smullyn. Without their influences,
this would have been a very different possible
world.*

MF

Preface

What's in This Book

After a rocky start in the first half of the twentieth century, modal logic hit its stride in the second half. The introduction of possible world semantics around the mid-century mark made the difference. Possible world semantics provided a technical device with intuitive appeal, and almost overnight the subject became something people felt they understood, rightly or wrongly. Today there are many books that deal with modal logic. But with only a small number of exceptions, treatments are almost entirely of *propositional* modal logic. By now this is a well-worked area and a standard part of philosophical training. *First-order* modal logic, on the other hand, is under-represented in the literature. It is not simply propositional modal logic plus quantifiers. Complications arise that have no counterpart in classical logic. The results are nuanced and expressive. That is what this book is about.

To make our book self-contained, we incorporate a refresher for those who need it, on classical propositional logic both axiomatically and with tableau systems, and with detailed proofs of soundness and completeness. Then we move on to modal logics. We begin with propositional modal logic, and our presentation here is typical of the entire book. Our basic approach is semantic, using possible world, or Kripke, models. Proof-theoretically, our primary machinery is semantic tableaux, which are easy and intuitive to use. We also include a treatment of propositional axiom systems, though axiomatics do not continue throughout the book. Philosophically, we discuss the issues that motivate formal modal logics and consider what bearing technical developments have on well-known philosophical problems. This three-pronged approach, Kripke semantics, tableaux, and philosophical discussion, reappears as each new topic is introduced.

Classically, first-order issues like constant and function symbols, equality, quantification, and definite descriptions, have straightforward formal treatments that have been standard items for a long time. Modally, each of these items needs rethinking. As noted above, first-order modal logic, most decidedly, is not just propositional modal logic plus classical quantifier machinery. The situation is much subtler than

that. Indeed, many of the philosophical problems attending the development of modal logic have arisen because of the myopia of looking at it through classical lenses. For this reason, we initially develop first-order modal logic without constants and functions. This arrangement separates the philosophical problems due to names and definite descriptions from those of quantifiers, and facilitates the development and understanding of first-order modal notions.

Successive chapters are arranged along the following lines. First, as we said, we introduce the quantifiers and address the difference between actualist quantification (where quantifiers range over what actually exists) and possibilist quantification (where they range over what might exist). Semantically, this is the difference between varying domain models and constant domain models. (If you don't know what this means, you will.) Next, we introduce equality. We discuss Frege's well-known morning star/evening star puzzle in this modal context, while presenting a formal treatment of equality. We then return to the matter of actualist and possibilist quantification and show how to embed possibilist quantification in an actualist framework and conversely.

By this point in the exposition, we are dealing with both a defined notion of existence and a primitive notion, and the time is ripe for a full-fledged discussion of the logical problems with existence and, in particular, of the idea that there are things that could exist but don't. Only after all of this has been examined are constants and function names introduced, along with the idea of predicate abstraction and rigidity.

Predicate abstraction is perhaps the most central notion of our treatment. This is a syntactic mechanism for abstracting a predicate from a formula, in effect providing a scoping mechanism for constants and function symbols similar to that provided for variables by quantifiers. Using predicate abstraction, we are able to deepen the understanding of singular terms and reference, enabling distinctions that were either not visible or only dimly visible classically. Since the variable binding of predicate abstracts behaves similarly to the variable binding of quantifiers, analogues to some of the issues involving Barcan and converse Barcan formulas crop up again.

We continue our discussion of nonexistence by distinguishing it from nondesignation: we contrast terms that designate a nonexistent, e.g., "the golden mountain," and terms that fail to designate at all, e.g., "the round square." In the final chapter, we present formal machinery for definite descriptions that borrows from both the Fregean and Russellian paradigms, and builds on the work in earlier chapters concerning these issues of existence and designation.

A primary goal of this text is to present first-order modal logic in an intuitively appealing way, so as to make more widely available versatile machinery for a new perspective on issues in philosophy of language and in metaphysics. Among other things, modal logic is a logic of change. It is our hope that this text on first-order modal logic will be a helpful agent of change, sweeping through philosophical discussions, bringing a freshness, liveliness, and new vigor to ancient issues.

How to Read This Book

We have two kinds of suggestions concerning your reading of this book: about where to start, and about what to skip.

This is a book on first-order modal logic. In principle it starts with Part III, specifically Chap. 8. If you are sufficiently familiar with *propositional* modal logic, then you might start here. However, a particular family of tableau proof systems is used throughout our modal discussion and, even if you are comfortable with modal semantics, you may wish to begin by reading some of the coverage of propositional modal tableaus in Part II. Also if you are relatively new to the philosophical ideas and formal treatment of modal logic even in the propositional setting, but have a good background with classical propositional logic, feel free to begin with Part II.

And finally, if you are new to, or uncomfortable with, formal proof procedures, soundness and completeness proofs, and the kind of mathematics appropriate for proving facts about such things, it would be good to start at the beginning, with Part I which, you will note, is labeled *Background*.

So much for where to start. In summary form, start where you feel comfortable starting. If you get into trouble, start again, but earlier.

Moving on to our second topic, you might wonder why we would suggest skipping anything. But we have our reasons.

The central idea of this book is the application of formal, mathematical methods to model and help elucidate some of the philosophical concepts that arise when modalities are studied. This begins with the basic notions of necessity and possibility themselves, which are formally modeled using possible world semantics, Kripke models. The book then proceeds to consider quantification, of more than one kind. It moves through predicate abstraction, considers rigid and non-rigid designators, designation and non-designation, several other things as well, and winds up with a study of definite descriptions in a modal setting.

For each of the items mentioned above, interest stems from philosophical considerations. Our comments on the philosophical motivations should be read and understood. Then formal methods are introduced that model some of these philosophical ideas. These involve formal semantics and formal proof procedures. That these exist and can help with our understanding is, basically, the central point of our book. All this should be read and internalized.

One can, with relatively minimal mathematical background, learn to use the semantical machinery presented here, and to use the formal proof procedures, primarily tableau systems. But, what is the relationship between semantics and proofs? They ought to be complementary. That is, the discovery of a tableau proof should be enough to stop us from looking for a semantic counter-model because it would be a waste of time. The discovery of a semantic counter-model to something should likewise stop us from wasting our time trying to construct a tableau proof. Semantics and proof methods should fit together.

As a matter of fact, our proof methods and our semantics do fit together. But establishing this requires mathematical arguments. As this book progresses, our

logics become more and more complicated, and proving that we do have this fit becomes harder and harder. We suggest that on a first pass you concentrate on understanding what the machinery of formal proofs and models is, how to use it, and how it has been designed to represent various philosophical ideas and positions. For this, it is enough to know that formal proofs and formal semantics do fit together. Once you are comfortable with this and the applications of this, then look at the mathematical proofs that semantics and tableaux fit together as we say they do.

Proofs that things fit together as just discussed are called proofs of *soundness* and of *completeness*. Such proofs occur throughout our book. As we go on, soundness and completeness proofs for each new version of things build on the corresponding proofs for the simpler versions that preceded them. Thus, work need not be repeated every time we start a new topic. But complexity grows as the machinery becomes more expressive.

Again to finish with a summary version, read first with the aim of understanding the machinery introduced, and how it can be used to elucidate various philosophical positions. Skip over the parts that are heavily mathematical. When comfortable with your understanding, go back and study the mathematical arguments that show it all works the way we said it did.

Differences from the First Edition

This is the second edition of our book. The philosophical discussions, history, and collection of detailed examples have all been expanded. But the most noticeable changes are technical. An introductory part concerning classical propositional logic has been added. While readers of this book doubtless know much about this topic already, it allows us to present the methodology of completeness and soundness proofs in a basic setting, before modal and quantificational complexities are added. In addition, we are able to present a tableau system in its simplest form, for those unfamiliar with such things. The material may be studied or skimmed depending on the background of the reader.

The machinery of predicate abstracts has been modified from the first edition. For instance, our tableau rules now can introduce only one non-rigid constant or function symbol in a step, rather than more complex terms as was earlier the case. This makes proof structure simpler in many ways. Some things that previously counted as legal formulas no longer do so, though only readers already very familiar with the first edition are likely to notice this difference. More obviously, we now explicitly present four different systems of logic using predicate abstracts, determined by whether quantifier domains are constant or not, and whether non-rigid terms must always denote or might fail to do so sometimes. These systems were actually present in the first edition, but this was not brought out clearly. Now it is made quite explicit, including detailed examinations of the changes needed in tableau rules.

The most significant technical difference between the present and the first edition is in the treatment of definite descriptions. The first edition presented tableau

rules for definite descriptions that were awkward and, if you will, un-tableau like. Each step in a tableau proof is, loosely speaking, supposed to introduce something simpler than what it derives from. The original tableau rules violated this, in effect smuggling in Russell's characterization (via Hintikka), and doing so in a way that was essentially axiomatic, and not in the spirit of tableaux. All this has been scrapped and new, and we believe simpler, tableau rules for definite descriptions are now featured.

Other than the changes mentioned above, all the material from the first edition is still in the present version. Things have been moved around, things have been added, but nothing essential has been lost.

New York, NY, USA

Melvin Fitting
Richard L. Mendelsohn

Acknowledgments

We wish to thank the following people who supplied corrections for and commentary on the first edition of this book: Mark Alfano, Thomas Bolander, Evgeny Borisov, Luke Hunsberger, Marko Mahling, Robert Mössgen, Dilip Ninan, Peter Steinacker, and Jack Woods. And we want to thank Andrzej Indrzejczak, C. L. Johnson, Nils Kürbis, Davide Leonessi, Toby Meadows, Eugenio Orlandelli, Niko Strobach, Branden Ward, Kai Wehmeier, Maya Von Ziegesar, and the members of the UCI Logic Seminar, for helpful comments on a draft of the present edition. We especially thank Dmitry Shkatov who read almost all of and Valentin Goranko who read the entire of a near final version of our book and commented extensively, improving it substantially.

Our greatest debt is, of course, to Saul Kripke, who profoundly reshaped logic and philosophy through his ground-breaking work on modal logic. Saul unfortunately passed away at the time we were working on this manuscript. It is our hope that new generations who study our book will become familiar with his work and, in so doing, keep his spirit alive.

Contents

Part I Background: Propositional Classical Logic

1	Background: Propositional Language	3
1.1	Introduction	3
1.2	The Propositional Language	4
1.3	Using Induction	5
2	Background: Propositional Axiomatics	9
2.1	Truth Tables	9
2.2	Axiom Systems	12
2.3	The Goal and General Outline	14
2.4	Consistency and Lindenbaum’s Lemma	16
2.5	Implication and the Deduction Theorem	19
2.6	The Other Connectives	24
2.6.1	Conjunction	24
2.6.2	Disjunction	25
2.6.3	Negation	26
2.6.4	Implication	26
2.7	Summary of Our Classical Axiom System	27
2.8	Completeness At Last	28
2.9	Redefining Consistency	29
3	Background: Propositional Tableaus	31
3.1	Tableaus	32
3.2	Logical Consequence Using Tableaus	35
3.3	Tableau Soundness	36
3.4	Tableau Completeness	38
3.4.1	Hintikka Sets	38
3.4.2	Completeness, Constructively	40
3.4.3	Tableau Completeness, Non-constructively	43
3.4.4	Coda	45
3.5	Strong Completeness and Compactness	46
	References	48

Part II Propositional Modal Logic

4	Modal Logic, an Introduction	51
4.1	What Is a Modal?	52
4.2	Can There Be a Modal Logic?	53
4.3	Aristotle's Modal Square	55
4.4	Informal Interpretations	56
4.5	Temporal Interpretations	59
4.6	Historical Highlights	63
4.6.1	Aristotle's Development of the Square	63
4.6.2	Aristotle's Future Sea Battle	65
4.6.3	The Master Argument of Diodorus Cronus	68
4.6.4	The Once and Future Conditional	71
4.6.5	The Reality of Necessity	72
	References	75
5	Propositional Modal Logic	77
5.1	What Are the Formulas?	77
5.2	What Are the Models?	78
5.3	Examples	81
5.4	Modal Logics, Semantically Defined	84
5.5	The Modal Cube	88
5.6	Semantic Consequence	90
	References	93
6	Propositional Modal Axiom Systems	95
6.1	The Logic $\mathcal{K}$ Axiomatically	95
6.2	More Axiom Systems	98
6.3	Logical Consequence, Axiomatically	101
6.4	Axiom Systems Work	103
6.4.1	Soundness	103
6.4.2	Completeness	104
6.5	Informal Notes	110
6.5.1	Gödel's Intuitionistic Logic Interpretation	110
6.5.2	Epistemic Logic	112
6.5.3	The Knowability Paradox	116
6.6	Justification Logic	118
	References	119
7	Propositional Modal Tableaus	121
7.1	Tableaus	121
7.2	More Tableau Systems	125
7.3	Logical Consequence and Tableaus	130
7.4	Modal Tableau Soundness	131
7.5	Modal Hintikka Sets	134

7.6	Propositional Modal Tableau Completeness	139
7.6.1	Modal Tableau Completeness, Constructively	140
7.6.2	Logical Consequence	143
7.6.3	Modal Completeness, Non-constructively	144
7.7	Other Kinds of Tableaus	148
7.7.1	Priest Style Tableaus	149
7.7.2	Negri Style Tableaus	151
7.7.3	Hybrid Logic Tableaus	153
	References	156

Part III First-Order Modal Logic

8	Quantified Modal Logic	159
8.1	First-Order Modal Formulas	159
8.2	An Informal Introduction	162
8.3	Necessity De Re and De Dicto	164
8.4	Is Quantified Modal Logic Possible?	168
8.5	What the Quantifiers Quantify Over	171
8.6	Constant Domain Models	173
8.7	Varying Domain Models	179
8.8	Free Logic, Briefly	183
8.9	Different Media, Same Message	184
8.10	Barcan and Converse Barcan Formulas	186
	References	193
9	First-Order Modal Tableaus	195
9.1	Constant Domain Modal Tableaus	196
9.2	Varying Domain Tableaus	199
9.3	Varying Domain Tableau Soundness	201
9.4	Hintikka Sets (Again)	203
9.5	Tableau Completeness with Quantification	205
9.6	The Completeness Proof, an Example	210
9.7	Completeness Using Maximal Consistency	212
	References	214
10	First-Order Modal Axiomatics	215
10.1	A Classical First-Order Axiom System	216
10.2	So What Are the Problems?	217
10.3	Constant Domain Systems	219
10.4	Varying Domain Systems	221
	References	223

Part IV Equality and Existence

11	Equality	227
11.1	Classical Background	227
11.2	Frege's Puzzle	230

11.3	The Indiscernibility of Identicals	232
11.4	The Formal Details	236
11.5	Tableau Equality Rules	237
11.6	Tableau Soundness with Equality	241
11.7	Hintikka Sets with Equality	242
11.8	Tableau Completeness with Equality	248
11.9	An Example	249
	References	252
12	Existence	253
12.1	To Be	254
12.2	Tableau Proofs	255
12.3	The Paradox of NonBeing	257
12.4	Deflationists	259
12.5	Parmenides' Principle	262
12.6	Inflationists	263
12.7	Unactualized Possibles	266
12.8	Barcan and Converse Barcan, Again	268
12.9	Using Validities in Tableaus	270
12.10	Tableaus Imitate Tableaus	272
12.11	On Symmetry	275
	References	276
 Part V Predicate Abstraction and Scope		
13	Predicate Abstraction, Informally	281
13.1	Why Constants Should Not Be Constant	281
13.2	Scope	285
13.3	The De Re/De Dicto Distinction, More Examples	287
13.4	The De Re/De Dicto Distinction: History	290
13.5	Understanding the Distinction: Possible Worlds and Scope	292
13.6	Predicate Abstraction: Informal Discussion	295
13.7	The Scope Distinction and Predicate Abstraction: Informal Discussion	296
13.8	Reading Predicate Abstracts	299
13.9	Actuality	301
13.10	What Is the Actuality Operator?	303
	References	305
14	Predicate Abstraction, Formally	307
14.1	Constant Symbol Syntax	308
14.2	Constant Symbol Semantics, Always Designate Case	309
14.3	Function Symbol Syntax	312
14.4	Function Symbol Semantics, Always Designate Case	314
14.5	Partiality and Designation	320
14.6	Non-Designation Formally	321

14.7	What We Still Can't Say	327
14.8	Extending The Notation to Predicates	328
	References	330
15	Tableaus for Predicate Abstraction	331
15.1	Quantification and Non-rigidity	331
15.2	Object Terms Syntactically	332
15.3	Being on a Branch	334
15.4	Constant Domain Tableau Rules	335
15.4.1	Constant Domains Assuming Terms Always Designate, <i>CA</i>	336
15.4.2	Constant Domains Assuming Terms Might Not Designate, <i>CN</i>	341
15.5	Varying Domain Tableau Rules	342
15.5.1	Varying Domains, Assuming Terms Always Designate, <i>VA</i>	343
15.5.2	Varying Domains, Assuming Terms Might Not Designate, <i>VN</i>	347
	Reference	349
16	Tableau Soundness and Completeness	351
16.1	Object Terms Semantically	351
16.2	A Technical Result	356
16.3	Soundness for K (<i>VN</i> Version)	358
16.3.1	Predicate Abstraction	359
16.4	Hintikka Sets for K (<i>VN</i> Version)	362
16.5	Adding A Witness World	365
16.5.1	Construction of H^w from H	366
16.6	Hintikka's Lemma, First Pass	367
16.7	Hintikka's Lemma, Second Pass	370
16.8	Completeness for K (<i>VN</i> Version)	372
Part VI Applications		
17	Equality and Predicate Abstraction	377
17.1	The Role of Scope	377
17.2	More Examples	380
18	Designation	383
18.1	Designation and Existence	383
18.2	Existence and Designation	386
	Reference	391
19	Rigidity	393
19.1	Rigid Designators	394
19.2	Rigidity Formally	396
19.3	Rigidity That Isn't Strong	402

19.4	A Dynamic Logic Example	404
	References	406
20	Definite Descriptions	407
20.1	Two Theories of Descriptions	407
20.2	Definite Description Syntax	411
20.3	Semantics for Definite Descriptions	413
20.4	Some Examples	414
20.5	Russell's Approach	422
20.6	Our Strong Recommendations	425
20.7	Tableaus for Definite Descriptions	426
	20.7.1 On Origins	430
20.8	Tableau Examples	431
20.9	Soundness, With Definite Descriptions	435
20.10	Hintikka Sets	438
20.11	Hintikka's Lemma	440
20.12	Completeness	446
	References	447
	Afterword	449
	Index	451

Part I

Background: Propositional Classical Logic

The overall structure of this book consists in the presentation of a sequence of formal logics whose expressiveness and complexity increase as we progress. For each logic a semantics will be given, as will formal proof procedures, and soundness and completeness results will be supplied. Of course applications to philosophical issues will be discussed at length since, after all, these applications are the primary reason for introducing the formal machinery. The introduction of machinery for each logic assumes that the material from previous and simpler logics that have been introduced in this book is known and understood, and so we do not need to repeat our work every time a new logical mechanism appears. We start this process here, with *propositional classical logic*, the basis of all that follows.

We assume readers are generally familiar with propositional classical logic, have seen truth tables, know how to read formulas, and so on. What we present is the prototype for the material that follows. For this logic we give a formal semantics and an axiomatic proof system, and then we prove soundness and completeness. We also give a tableau system for the logic, and prove soundness and completeness for this. Thus we have three ways of characterizing classical tautologies: semantically, axiomatically, and via tableaux, all of which have been proved to be equivalent. This is the pattern throughout the book, except that from a certain point on axiomatics is dropped. This is not because it loses interest, but because it becomes complex in ways that are not particularly relevant to the themes of the present book.

Different readers will come to this book with different backgrounds. This Part of the book is for those who are relatively new to formal methods. Readers who are more experienced may wish to skip all this and begin with Part II. One central feature of our approach is the use of tableau systems of proof, and some readers may wish to read the material on these in the classical setting where they are simplest, and skip the rest of this Part. And some may feel most comfortable simply beginning at the beginning. Please feel free to do what you consider best for you.

Chapter 1

Background: Propositional Language



1.1 Introduction

Logics are generally specified in two fundamentally different ways, using proofs and using models. Both are important, and the two are intimately connected.

Informally, a proof is an argument that convinces. Formally, a proof is of a particular formula, and is a finite object constructed according to syntactic rules that refer only to the structure of formulas and not to their intended meaning. The syntactic rules that define proofs are said to specify a *proof procedure*. Generally, one shows a formula is a *theorem* by presenting a proof for it. But this is only part of the story. How might one show some formula is not a theorem—that it has no proof? Finding a flaw in a proof does not show the formula in question is not provable, since a different proof might actually establish the result.¹

The other way of specifying a logic is *semantically*. One introduces a class of *models* with rules for evaluating the truth of formulas in a model. A formula that evaluates to true in every model for the logic is called *valid* in the logic. Commonly the class of models for a logic can be infinite, which makes simple checking of all the cases impossible. But semantical methods are often used in reverse. To show a formula is not valid, producing a single counter-model suffices.

Ideally a logic will have a proof procedure and a semantics that fit together, so that the provable theorems and the semantical validities are the same formulas. If this happens, then giving a counter-model for a formula shows no proof exists, and giving a proof shows no counter-model can be found. We then have the possibility

¹ In 1858, Abraham Lincoln closed his part of the Charleston Lincoln-Douglas debate with the following observation. “If you have ever studied geometry, you remember that by a course of reasoning Euclid proves that all the angles in a triangle are equal to two right angles. Euclid has shown you how to work it out. Now, if you undertake to disprove that proposition, and to show that it is erroneous, would you prove it to be false by calling Euclid a liar?” Taken from *The Complete Lincoln-Douglas Debates of 1858*, edited by P. M. Adams, University of Chicago Press, 1991.

of producing an actual witness for validity (a proof) and an actual witness for non-validity (a counter-model). This will be the case for all logics considered in this book.

We will see that there are proof procedures for a range of different logics that all have certain family resemblances, and similarly for their semantics. What is called *classical propositional logic* is the simplest common case, and we have chosen to start with it. This allows us to present certain basic ideas that will turn up over and over for more complex logics. We will be able to see the important features with a minimum of machinery. As we said earlier, if you are already comfortable with classical propositional logic you could skip ahead, or browse quickly through to refresh your memory.

There are many kinds of proof procedures, divided into two broad categories: *analytic*, and those that are not (sometimes called *synthetic*). Analytic proof procedures only make use of subformulas of the formula being proved (or some related, but restricted, set of formulas). This makes it easier to find proofs, and gives insights that one might not otherwise have had. We will see a non-analytic proof procedure for classical propositional logic in Chap. 2, an axiom system. And we will see an analytic proof procedure, tableaux, in Chap. 3.

Something that generally remains hidden in the background: classical propositional logic is not actually a thing, but a family. Indeed, this is actually true of most logics. Syntactically, we can assume we have a large variety of basic propositional connectives, or a small set with others defined from them. Axiomatically, a wide variety of axiom choices are to be found in the literature. Many other kinds of proof procedures can be used in place of an axiom system. Ultimately such details don't matter much, because for each version in common use, all the others can be embedded, or translated, into it. Thus the differences can be, and commonly are, ignored. The term *classical logic* usually refers ambiguously to any of these. When it matters, one specifies. When it does not, one ignores the issue. But when starting out, some choice must be made. We do so below, with the understanding that as the book goes on, the exact details of our choice diminish in importance.

1.2 The Propositional Language

The language of classical propositional logic is the simplest of any of the formal languages that will be used in this book. Still, the machinery for working with it will reappear in various modifications throughout. It is best to see basic simplicity before complexity shows up.

We start with a countable list of *propositional letters*, $P_1, P_2, \dots$, sometimes called *propositional atoms* or *atomic formulas*. Formulas will be built up from these using propositional connectives, and here is one of those places where a range of choices is possible. There is, however, one guiding principle behind a choice of connectives. It is possible semantically to define two place logical operators, three place, and so on. We need to choose enough as basic so that all the rest are all

definable. Details of this are skipped here. Suffice it to say, our choice meets this condition. It is not minimal, because we want to include the connectives in most common use. We will take as primitive certain *binary connectives*. specifically: *material implication*, $\supset$; *disjunction*, $\vee$; and *conjunction*, $\wedge$. We also take as primitive the *unary connective* of *negation*, $\neg$.

Given all this, here is the definition of the set of propositional formulas. It is a recursive definition, and says how formulas are generated.

Definition 1.2.1 (Classical Propositional Formula) The set of formulas is the smallest set meeting the following conditions:

1. Every propositional atom is a formula.
2. If X and Y are formulas, then $(X \wedge Y)$, $(X \vee Y)$ and $(X \supset Y)$ are formulas.
3. If X is a formula, then so is $\neg X$.

We will sometimes make use of material equivalence, $\equiv$, but for us it is easier to take it as defined instead of primitive: from here on $X \equiv Y$ abbreviates $((X \supset Y) \wedge (Y \supset X))$.

Using the definition it is easy to show that, say, $((P \vee Q) \supset (P \wedge Q))$ is a formula provided P and Q are propositional atoms. It is less easy to show that $((P \vee Q) \supset (P \wedge Q))$ is not a formula (note that there is a parenthesis missing). Devising tests for formula-hood is something that is best done using automata theory. Here we'll assume you can recognize formulas when you see them. Further, we will often be somewhat informal and omit outer parentheses when convenient, writing $(X \vee Y) \supset (X \wedge Y)$ for $((X \vee Y) \supset (X \wedge Y))$, for instance. We may occasionally write square or curly brackets in place of parenthesis, as an aid in reading formulas. These are informal conventions, with no formal significance here.

1.3 Using Induction

Propositional formulas constitute a particular class of words. The clauses in Definition 1.2.1 amount to rules for generating the members of this class, formulas. There is a ground case, 1, to get things started. And there are recursive cases, 2 and 3, telling us how to create new formulas from ones we have already generated. These generative cases are what is called *positive*. They say certain things are formulas provided certain other things *are* formulas. There are no clauses of a form saying something is a formula provided certain other things *are not* formulas, negative conditions. Whenever we have a class of words defined with positive generating clauses of this sort, it can be shown that a unique class is always determined. All our formula definitions will be like this.

Sometimes we will need to prove things about the class of formulas. For this, mathematical induction is a useful tool. You probably are familiar with induction in the following form. Let us take *numbers* to mean $0, 1, 2, \dots$. Mathematical induction says, if 0 has some property $\mathcal{P}$, and if $n + 1$ has property $\mathcal{P}$ whenever n does, then

every number has property $\mathcal{P}$. But you may or may not be aware of the following alternate version.

Proposition 1.3.1 (Complete Induction) *Suppose a number n has property $\mathcal{P}$ whenever every smaller number does. Then every number has property $\mathcal{P}$.*

If you haven't seen Complete Induction before, here is the intuition. Every member of the empty set has property $\mathcal{P}$ (show me one that doesn't). But then every number smaller than 0 has $\mathcal{P}$, so 0 must have $\mathcal{P}$. And then every number smaller than 1 has property $\mathcal{P}$ so 1 does. But then 0 and 1 have $\mathcal{P}$ so $\dots$ Complete Induction can, in fact, be derived from the ordinary version of induction, but we do not go through this here. We simply assume Complete Induction as a mathematical fact.

In order to use Complete Induction to prove things about formulas, we need to get numbers into it, and for this we make use of the following standard bit of logical terminology.

Definition 1.3.2 (Degree) The degree of a formula is the total number of occurrences of $\neg$, $\wedge$, $\vee$, and $\supset$ in it.

Note that propositional letters have degree 0. Now, the key fact: the degree of any immediate subformula of a formula is less than the degree of the formula itself. (We have not defined these terms, but they should be fairly obvious.) For instance, the degree of X and the degree of Y must be less than the degree of $(X \wedge Y)$. The degree of $(X \wedge Y)$ is the degree of X plus the degree of Y plus 1, because of the occurrence of $\wedge$. Thus the degree of $(X \wedge Y)$ is at least 1 greater than the degree of X , though it could be more depending on Y . (The reason we want Complete Induction and not the ordinary version of induction is because degrees need not go up by exactly 1 as formulas are formed, as we see in this case.)

Here is an example of using Complete Induction to prove something about all formulas: we show every formula has the same number of left and right parentheses. Just this once we are a bit elaborate in our proof presentation so you see how this works. In the future we will be more succinct.

Let property $\mathcal{P}$ be the property that holds of number n just in case all formulas of degree n have the same number of left and right parentheses. If we show every number has property $\mathcal{P}$, that will show every formula has the same number of left and right parentheses.

Suppose F is a formula, and it is known that every formula of lower degree has the same number of left and right parentheses. We show this is true of F itself. There are five cases: F is atomic, a negation, a conjunction, a disjunction, or an implication.

If F is atomic, there are no formulas of lower degree, so the induction hypothesis is vacuously true. But atomic formulas do have the same number of left and right parentheses, namely 0, and no induction hypothesis was actually needed.

Suppose F is a conjunction, say $(X \wedge Y)$. Then both X and Y are of lower degree, so the induction hypothesis applies and each has the same number of left and right parentheses. But then the number of left parentheses in F is the number in X plus

the number in Y plus 1, and by the induction hypothesis this is the number of right parentheses in X plus the number of right parentheses in Y plus 1, and this is the number of right parentheses in $(X \wedge Y)$, or F .

The other cases are clearly similar, and we leave them out.

Incidentally, this now gives us an easy way of proving that $((P \vee Q \supset (P \wedge Q))$ is not a formula. It has more left parentheses than right.

Exercises

Exercise 1.3.1 Find a way of *proving* $((P \vee Q) \supset P \wedge Q)$ is not a formula.

Chapter 2

Background: Propositional Axiomatics



In this chapter we construct an axiomatic proof system for classical propositional logic. In the next chapter we revisit the logic, but with semantic tableaux as the main proof method. There are many logics in use today but all have a certain commonality. Syntactically there is a specification of a formal language. There is some notion of a semantics, providing a mathematically defined *meaning* for formulas of the language. There is some specification of a proof system. And finally, there are connections established between the semantics and the proof system. All this is at its clearest and simplest for classical propositional logic. It is the most well-behaved of all the logics. Think of our presentation in this and the next chapter as providing a foundation on which many more elaborate logics can be built. Our interest, of course, will be on those that are modal.

2.1 Truth Tables

Formulas are pure syntax. They have to be assigned some meaning, and this is the job of semantics. For classical propositional logic, essentially, the meaning of a formula is simply truth or falsehood. Truth tables are the standard tool for this—each line represents a context, specified by assigning values to propositional letters, and the truth value calculated for a formula on a truth table line constitutes the meaning of the formula in that context. Almost certainly you all have seen truth tables, and we won't go into their details. What we will do is extract their mathematical essence, because it will be convenient later on.

Let us assume we have two *truth values*, **t** and **f**. (Exactly what these are is not important here, only that they are different from each other.) The intuition, of course, is that they represent truth and falsehood. We assume there are various operations defined on the set $\{\mathbf{t}, \mathbf{f}\}$ of truth values, corresponding to our choice of connectives. For convenience, we use $\wedge$ as both a connective, a formal symbol, and as the name of

an operation on truth values, and similarly for the other connectives. These are really quite different things, but you can tell from context which is meant. It is analogous to the situation one encounters in mathematics, where one sees similar ambiguity. On the one hand we might talk about the leftmost occurrence of the symbol $+$ in the equation $(x + y) = (y + x)$, and here we are using syntax. Or we might say $3 + 5$ is the number 8, here we are applying an operation. While these are quite different, there is usually no problem keeping the differences sorted out.

Here are definitions of one unary operation on the set $\{\mathbf{t}, \mathbf{f}\}$ of truth values, and three binary operations.

X	$\neg X$	X	Y	$X \supset Y$	$X \wedge Y$	$X \vee Y$
$\mathbf{t}$	$\mathbf{f}$	$\mathbf{t}$	$\mathbf{t}$	$\mathbf{t}$	$\mathbf{t}$	$\mathbf{t}$
$\mathbf{t}$	$\mathbf{f}$	$\mathbf{t}$	$\mathbf{f}$	$\mathbf{f}$	$\mathbf{f}$	$\mathbf{t}$
$\mathbf{f}$	$\mathbf{t}$	$\mathbf{f}$	$\mathbf{t}$	$\mathbf{t}$	$\mathbf{f}$	$\mathbf{t}$
$\mathbf{f}$	$\mathbf{t}$	$\mathbf{f}$	$\mathbf{f}$	$\mathbf{t}$	$\mathbf{f}$	$\mathbf{f}$

To connect these operations with formulas, we have the following.

Definition 2.1.1 (Boolean Valuation) A *boolean valuation* is a mapping v from the set of formulas to the set of truth values that meets the following conditions:

1. $v(\neg X) = \neg v(X)$
2. $v(X \supset Y) = v(X) \supset v(Y)$
3. $v(X \wedge Y) = v(X) \wedge v(Y)$
4. $v(X \vee Y) = v(X) \vee v(Y)$

Take a typical line of this definition, say $v(X \supset Y) = v(X) \supset v(Y)$. The occurrence of “ $\supset$ ” on the left is syntactical—it is part of the formula $(X \supset Y)$. The occurrence on the right is the operation from the table above.

It can be shown that any two boolean valuations that agree on the propositional letters of a formula will agree on the formula. It follows that the value of a boolean valuation v on a formula X is not affected by changing v on propositional letters that don’t occur in X . It can also be shown that a boolean valuation is completely specified by giving its values on propositional letters. We’ll assume all this here, and leave some of it to you as an exercise for later.

As an example, suppose $v(P_1) = \mathbf{t}$ and $v(P_2) = \mathbf{f}$. We compute $v(P_1 \supset (P_2 \vee (\neg P_1 \wedge P_2)))$.

$$\begin{aligned}
 v(P_1 \supset (P_2 \vee (\neg P_1 \wedge P_2))) &= v(P_1) \supset v(P_2 \vee (\neg P_1 \wedge P_2)) \\
 &= v(P_1) \supset (v(P_2) \vee v(\neg P_1 \wedge P_2)) \\
 &= v(P_1) \supset (v(P_2) \vee (v(\neg P_1) \wedge v(P_2))) \\
 &= v(P_1) \supset (v(P_2) \vee (\neg v(P_1) \wedge v(P_2))) \\
 &= \mathbf{t} \supset (\mathbf{f} \vee (\neg \mathbf{t} \wedge \mathbf{f}))
 \end{aligned}$$

$$\begin{aligned}
&= \mathbf{t} \supset (\mathbf{f} \vee (\mathbf{f} \wedge \mathbf{f})) \\
&= \mathbf{t} \supset (\mathbf{f} \vee \mathbf{f}) \\
&= \mathbf{t} \supset \mathbf{f} \\
&= \mathbf{f}
\end{aligned}$$

You should recognize that the calculation above simply embodies all the steps involved in filling in a standard truth table line for $v(P_1 \supset (P_2 \vee (\neg P_1 \wedge P_2)))$ where the line is the one that assigns P_1 the value $\mathbf{t}$ and P_2 the value $\mathbf{f}$. It is more convenient here for us to use boolean valuations, which are mathematical functions, rather than truth tables as such. A table can be thought of as the specification of a function.

Now we use the semantics just defined to characterize the fundamental semantic notions of classical propositional logic.

Definition 2.1.2 (Tautology) A formula X is a *tautology* if $v(X) = \mathbf{t}$ for every boolean valuation v .

Informally, this amounts to saying X is a tautology if every line of a truth table for X assigns to X the value $\mathbf{t}$. Being a tautology has a generalization to being a consequence of a set of formulas.

Definition 2.1.3 (Semantic Consequence) Let S be a set of formulas, and let X be a single formula. We say X is a *semantic consequence* of S just in case every boolean valuation that maps all members of S to $\mathbf{t}$ also maps X to $\mathbf{t}$. If X is a semantic consequence of S we may write this as $S \models X$.

Loosely, X is a semantic consequence of S if X must be true whenever all members of S are true. Notice that saying X is a tautology is equivalent to saying X is a semantic consequence of $\emptyset$. Also note that if X is a tautology, it is a semantic consequence of every set.

Exercises

Exercise 2.1.1 Let w be a boolean valuation such that $w(P_1) = w(P_3) = \mathbf{t}$ and $w(P_2) = \mathbf{f}$. Evaluate $w((P_1 \wedge P_3) \supset (P_2 \supset \neg P_1))$.

Exercise 2.1.2 Which of the following are tautologies?

1. $P \vee \neg P$
2. $(P \supset Q) \supset (\neg P \supset \neg Q)$
3. $(P \supset Q) \supset (\neg Q \supset \neg P)$

Exercise 2.1.3 Use Complete Induction and prove that any two boolean valuations that agree on the propositional letters of a formula will agree on the formula.

Exercise 2.1.4 Show how $\wedge$ and $\vee$ can be characterized using $\neg$ and $\supset$. That is, find a formula $\varphi(P, Q)$ built up from P and Q using only the connectives $\neg$ and $\supset$ such that $(P \wedge Q) \equiv \varphi(P, Q)$ is a tautology. Similarly for $P \vee Q$.

Exercise 2.1.5 For the notion of semantic consequence from Definition 2.1.3, show that if $S \models X$ and $S \subseteq S'$ then $S' \models X$.

2.2 Axiom Systems

Constructing a truth table for a formula with many propositional letters is a very large task. Nonetheless, constructing a truth table is a finite process, and so it is decidable whether or not a formula is a tautology. For more complex logics, the semantics for the logic may not provide an effective method for determining validity (for logics other than classical propositional, one uses the term *valid* instead of tautology). What is often used in place of a direct application of the semantics for the logic is some notion of formal proof. Speaking casually, a proof is a convincing argument. Formally, a proof is a finite object that acts as a certificate that a formula is a validity. Of course this is a loose characterization, but our discussion below should help make the idea more precise.

Axiom systems provide a very common proof methodology—an axiomatic proof can be a relatively small object compared to a truth table (though a proof may be hard to discover). In this section we set up the basics of the axiomatic approach, then in subsequent sections we work towards soundness and completeness (and explain what these terms mean). To have an axiom system two things are needed: we need some recognizable class of formulas called *axioms*, whose truth is simply assumed; and we need *rules of derivation*, for producing new truths from old, so to speak. A word about each of these.

Many sets of formulas could be taken as a set of axioms, but practicality issues come into it. If we take the set of all tautologies as axioms, all axiomatic proofs would be trivial, and nothing would be gained over truth tables. Or, we could take as a set of axioms some infinite set of formulas whose Gödel numbers (if you know what they are) form a non-recursive set. In this case we would have no general way of telling what is and what is not an axiom. Reasonable restrictions are needed.

It is very common to specify a set of axioms by giving a set of *axiom schemes*: any formula of such-and-such form is an axiom. For example, we might say that any formula of the form $(X \supset (Y \supset X))$ is an axiom. If we did this, among the axioms would be $(P_1 \supset (P_2 \supset P_1))$ and also $((P_1 \wedge P_2) \supset (P_3 \supset (P_1 \wedge P_2)))$. In this approach $(X \supset (Y \supset X))$ is an *axiom scheme*, and we have given examples of specific axioms that are *instances* of this scheme. This is the way we will do things here—we will specify axiom systems by giving a finite number of axiom schemes.

Rules of derivation must be finitary and effective. Finitary means we need a finite number of formulas to trigger a rule application. Effective means we can tell when a rule can be applied. In practice, our rules will be specified by writing something

like the following:

$$\frac{X_1 \ X_2 \ \dots \ X_n}{Y}$$

where this means that if we have specific formulas that match each of the n formulas above the line, we can conclude the formula that results, as shown below the line. Two familiar examples of rules are the following.

$$\text{Modus Ponens} \quad \frac{X \ (X \supset Y)}{Y}$$

$$\text{Conjunction} \quad \frac{X \ Y}{(X \wedge Y)}$$

Note that just as we are using axiom *schemes*, our rules of inference are also schematic. Thus *Modus Ponens* really says that if we have a formula, and another formula that is an implication with the first formula as its antecedent, then we may conclude the formula that is its consequent. It represents a pattern of inference.

Now, to get a little more specific (but not very much so yet). It turns out that quite a lot can be said about the behavior of an axiom system without knowing what the choice of axioms or rules of inference actually are. For the time being we work in a very general setting, and postpone our choice of axioms and rules until later on when we have some idea of what we will need.

In what follows, suppose $\mathcal{A}$ is a finite set of axiom schemes. We will say a formula is an *axiom of $\mathcal{A}$* if it is an instance of one of the schemes in $\mathcal{A}$. And suppose $\mathcal{R}$ is a finite set of rules of derivation. These two together determine notions of a derivation, and of a proof, as follows.

Definition 2.2.1 (Derivation, Consequence, Proof) Let S be a set of formulas (not schemes, and not necessarily finite). By a *derivation from S* , in the system with axiom schemes $\mathcal{A}$ and rules $\mathcal{R}$, we mean any finite sequence of formulas

$$\begin{array}{c} X_1 \\ X_2 \\ \vdots \\ X_n \end{array}$$

in which each formula is: an axiom of $\mathcal{A}$, or a member of the set S , or follows from earlier formulas in the sequence using a rule from $\mathcal{R}$. The last formula, X_n , is the formula that is derived.

If the set S is empty, we say the derivation is a *proof*, and the last formula is the formula that has been proved.

If X has a derivation from S , we symbolize this with $S \vdash X$, and say X is a *consequence* of S (with $\mathcal{A}$ and $\mathcal{R}$ understood). If X is provable, that is, if $\emptyset \vdash X$, we just write $\vdash X$.

Technically, an axiom system is a pair, $\langle \mathcal{A}, \mathcal{R} \rangle$, consisting of a set of axioms and a set of rules. It will commonly cause us no problems if we simply use $\mathcal{A}$ to refer to such an axiom system, leaving $\mathcal{R}$ to be understood. There are a few easy facts about these notions that can now be established.

Proposition 2.2.2 (Monotonicity) *If $S \vdash X$ and $S \subseteq S'$, then $S' \vdash X$.*

Proof If $S \subseteq S'$ then any derivation from S is also a derivation from S' .

Proposition 2.2.3 (Compactness) *If $S \vdash X$ then there is some finite subset S' of S such that $S' \vdash X$.*

Proof Suppose $S \vdash X$, and so we have a derivation of X from S . That derivation has only finitely many lines, and so only uses a finite number of members of S . Let S' be the set of members of S actually used in the derivation. Then S' is finite, and the same derivation of X from S is also a derivation of X from S' .

2.3 The Goal and General Outline

There are many axiom systems having a finite number of axiom schemes and rules, that can prove exactly the tautologies. More specifically, there are axiom systems for which we can prove *soundness* and *completeness*, which means this.

Definition 2.3.1 (Soundness and Completeness) A classical propositional axiom system is *sound* if it only proves tautologies. A classical propositional axiom system is *complete* if it proves every tautology.

Soundness and completeness are easy to manage separately. An axiom system with no axioms at all can't prove anything, and hence it is sound (find me something it proves that is not a tautology). It is not very useful since it is badly incomplete. An axiom system that proves every formula is obviously complete—it proves everything hence it proves all tautologies. It too is not very useful since it is badly unsound. We need the two conditions together. In fact, soundness is rather simple to manage axiomatically.

Proposition 2.3.2 (Soundness) *Suppose we have an axiom system that meets the following two conditions:*

1. *Every axiom is a tautology.*
2. *Every rule is sound, by which we mean that any boolean valuation that maps all the premises of a rule application to $\mathbf{t}$ must also map the conclusion of the rule to $\mathbf{t}$.*

Then the axiom system is sound; it only proves tautologies.

Proof It is easy to see that if the conditions are met, every line of a proof must be a tautology. Hence the last line, which is what the proof proves, must be a tautology.

In fact a stronger result can be proved, which we leave to you as Exercise 2.3.2.

When we come to actually choosing axioms and rules, we will make sure to choose axiom schemes all of whose instances are tautologies, and whose rules of derivation are all sound. This guarantees we have a sound axiom system. Completeness is much harder, however, and it will occupy the next several sections. Since it is more complex, here is something of a guide.

Suppose we have an axiom system that we wish to prove is complete. We must show that every tautology is provable. Actually, we will show the converse: if X is not provable, then X is not a tautology. To show X is not a tautology, we must find a boolean valuation v that maps X to **f**. Let us take a look, again, at how boolean valuations behave.

Let v be a boolean valuation, and consider the set of all formulas that v maps to **t**. What is this set like? Well, the following is easy to see. Such a set must contain $Z \wedge W$ exactly when it contains both Z and W , because the only way a boolean valuation can map $Z \wedge W$ to **t** is by mapping each of Z and W to **t**. Similarly, such a set must contain $Z \vee W$ exactly when it contains at least one of Z or W . It must contain exactly one of Z or $\neg Z$ for each formula Z . And finally, it must contain $X \supset Y$ exactly if either it does not contain X or it does contain Y . (This is more complicated than the other conditions, but a check with the truth table for implication shows it to be the case.) Suppose we could show that a formula X that is not provable in our axiom system must fail to be a member of some such set. This would establish completeness (in its contrapositive form) because if we had such a set omitting X , we could immediately construct a boolean valuation mapping X to **f**.

In the next section we will introduce enough machinery so that we can define a notion of axiomatic *consistency*. We will show that there are consistent sets that cannot be enlarged without becoming inconsistent—these are called *maximally* consistent sets. Then we will take as our guiding principle: choose our axioms so that maximally consistent sets must have the properties we discussed above. Then maximally consistent sets will serve as the appropriate tools to prove completeness.

Exercises

Exercise 2.3.1 Show that if S is defined as the set of formulas that a boolean valuation v maps to **t**, then S has all the conditions discussed above: exactly one of X and $\neg X$ must be in S , and so on.

Exercise 2.3.2 Suppose we have an axiom system for which the two conditions of Proposition 2.3.2 are met. Show that if $S \vdash X$ using this axiom system, then $S \models X$ (Definition 2.1.3).

2.4 Consistency and Lindenbaum's Lemma

We now prove a very general and fundamental result concerning the existence of maximal consistent sets, generally known as *Lindenbaum's Lemma*. Of course we need a notion of consistency before we talk about maximal consistency. There is more than one way of doing this. One might, for instance, say a set of formulas is consistent if it does not have a contradiction as a consequence, say $P \wedge \neg P$. But this requires some knowledge of how conjunction and negation behave, and as of yet we have not adopted any specific axioms, so in particular we have none concerning these operations. We have chosen to use a definition of consistency that allows us to get to work now and pick our axioms afterward. In the rest of this chapter, all of our development should be thought of as using an axiom system that we will call $\mathcal{A}$, with rules of inference $\mathcal{R}$, about which we initially have no specific knowledge. Later we will gradually fill in what the axioms and rules that we want are.

Definition 2.4.1 (Consistency) With respect to axiom schemes $\mathcal{A}$ and rules of inference $\mathcal{R}$, a set S of formulas is *inconsistent* if it has a finite subset that has every formula as a consequence. More formally, S is inconsistent if there is some finite $S_0 \subseteq S$ such that $S_0 \vdash X$ for every formula X . S is *consistent* if it is not inconsistent.

A finite subset of a set S is also a finite subset of any extension of S . It follows that any extension of an inconsistent set is also inconsistent. More formally stated, we have the following.

Proposition 2.4.2 *Suppose $S \subseteq S'$. If S is inconsistent, so is S' . Consequently if S' is consistent, so is S .*

It is common in mathematics to call something *maximal* if it cannot be made bigger. This applies to the notion of consistency.

Definition 2.4.3 (Maximal Consistent) A set S of formulas is *maximally consistent* if: (1) S is consistent but, (2) no proper extension of S is consistent.

Another very useful way of saying part (2) of the definition above is this: any proposed enlargement of S that is consistent is not really an enlargement. That is, if the result of adding X to S is consistent, then X was already in S . More formally, our working version of maximality for a consistent set S is:

for every formula X , if $S \cup \{X\}$ is consistent, then $X \in S$.

Maximally consistent sets have many nice and useful properties. For instance, they are closed under derivation.

Proposition 2.4.4 *Suppose S is a maximally consistent set using axiom system $\mathcal{A}$ and rules $\mathcal{R}$. If $S \vdash X$, then $X \in S$.*

Proof Assume S is maximally consistent, and $S \vdash X$. If we can show that $S \cup \{X\}$ is consistent, then $X \in S$ because of maximal consistency. So showing that $S \cup \{X\}$ is consistent is our goal. We do this by contradiction: we suppose that $S \cup \{X\}$ is not consistent and derive a contradiction, specifically that S itself would then be inconsistent.

By our supposition that $S \cup \{X\}$ is not consistent, there is some finite subset of it that has every formula as a consequence. This finite subset must include X , because otherwise S itself would be inconsistent, contrary to our assumption that it is maximally *consistent*. So, there must be a finite subset S_0 of S such that $S_0 \cup \{X\}$ has every formula as a consequence. Also $S \vdash X$ so by Proposition 2.2.3 there is some finite subset S' of S such that $S' \vdash X$. Then $S_0 \cup S'$ is a finite subset of S . We will show that it has every formula as a consequence, which tells us that S itself is inconsistent.

Let Z be an arbitrary formula. Then $S_0 \cup \{X\} \vdash Z$, so we have a derivation of Z from $S_0 \cup \{X\}$. We will show how to convert this derivation into one showing that $S_0 \cup S' \vdash Z$.

The conversion process is really very simple. First, we are assuming that there is a derivation showing that $S' \vdash X$, say it is the following.

$$\begin{array}{c} Y_1 \\ Y_2 \\ \vdots \\ Y_k = X \end{array}$$

Also since $S_0 \cup \{X\} \vdash Z$ we have a derivation showing this, say it is the following.

$$\begin{array}{c} Z_1 \\ Z_2 \\ \vdots \\ Z_n = Z \end{array}$$

We first simply combine these two derivations, as follows.

$$\begin{array}{c} Y_1 \\ Y_2 \\ \vdots \\ Y_k = X \\ Z_1 \\ Z_2 \\ \vdots \\ Z_n = Z \end{array}$$

Now, among the Z_i there may be occurrences of X , since this part allowed X as one of the premises. Get rid of these X occurrences—we have X present anyway, as line k in the first part. It is not hard to check that the result is a derivation from $S_0 \cup S'$, and it is a derivation of Z .

Corollary 2.4.5 *Again suppose S is a maximally consistent set. Then S contains every theorem.*

Proof Let X be a theorem. Then $\emptyset \vdash X$. Since $\emptyset \subseteq S$, we do have $S \vdash X$ so by Proposition 2.4.4, $X \in S$.

We have, indeed, established some nice properties of maximally consistent sets, but this is not enough, because it is not obvious that there are any maximally consistent sets. If there were none, they would all have every nice property you might dream of, and this would be of no use at all. Fortunately a famous result due to Lindenbaum says that if there are any consistent sets, then there are maximally consistent sets. Indeed, generally there are lots of them. Before we prove this, it will be useful to have the following available—it plays a central role in establishing Lindenbaum's result.

Proposition 2.4.6 (Chain Limit) *Suppose we have a sequence of sets of formulas, $S_1, S_2, S_3, \dots$, so that $S_1 \subseteq S_2 \subseteq S_3 \subseteq \dots$ (Such things are called chains.) Let $S_\infty = S_1 \cup S_2 \cup S_3 \cup \dots$. Then S_∞ is a kind of limit of the chain, and we have $S_1 \subseteq S_2 \subseteq S_3 \subseteq \dots \subseteq S_\infty$. We claim the following:*

S_∞ is consistent if and only if S_n is consistent for every n .

Proof The proof has two parts.

Part 1: Suppose S_∞ is consistent. Then S_n is consistent for each n , since $S_n \subseteq S_\infty$ and we have Proposition 2.4.2.

Part 2: Suppose S_∞ is not consistent. Then there is a finite subset $S \subseteq S_\infty$ such that S has every formula as a consequence. If we show that $S \subseteq S_n$ for some n , we will have a member of the chain that is not consistent. To keep the notational clutter down, let's say that S has three members—the general case follows the same idea. Say $S = \{Z_1, Z_2, Z_3\}$. Since $S \subseteq S_\infty$, $Z_1 \in S_\infty$, so by the definition of S_∞ , $Z_1 \in S_a$ for some integer a . Similarly $Z_2 \in S_b$ and $Z_3 \in S_c$ for some integers b and c . Let n be whichever of a, b, c is largest. Since we have a chain structure, all of Z_1, Z_2 , and Z_3 will be in S_n , that is, $S \subseteq S_n$ where $n = \max\{a, b, c\}$.

Now for the main event.

Theorem 2.4.7 (Lindenbaum's Lemma) *Let S be any consistent set. Then $S \subseteq S_\infty$ for some maximally consistent set S_∞ . Less formally, every consistent set can be extended to a maximally consistent one.*

Proof Assume for the following that S is a consistent set of formulas.

We have in our language a countable list of propositional letters. There is a basic result of set theory, that would take us too far out of our way to prove here, that

says we have the following consequence: the set of *all* formulas of our language is countable. This means it is possible (constructively, in fact), to set up an infinite list, $X_1, X_2, X_3, \dots$, in such a way that every formula appears somewhere in the list. Let us assume this has been done.

We now define a sequence of consistent sets, $S_1, S_2, S_3, \dots$, as follows.

To start things off, let $S_1 = S$. Since S is consistent by assumption, the sequence of sets starts with a consistent set.

Next, suppose S_n has been defined, and is consistent. We define the next set as follows.

$$S_{n+1} = \begin{cases} S_n \cup \{X_n\} & \text{if } S_n \cup \{X_n\} \text{ is consistent} \\ S_n & \text{if } S_n \cup \{X_n\} \text{ is not consistent} \end{cases}$$

Notice that in either case S_{n+1} must be consistent. Notice also that $S_n \subseteq S_{n+1}$. We thus have a chain structure, as in Proposition 2.4.6. Let $S_\infty = S_1 \cup S_2 \cup S_3 \cup \dots$. Then we have:

$$S_1 \subseteq S_2 \subseteq S_3 \subseteq \dots \subseteq S_\infty$$

It is obvious that S_∞ extends $S = S_1$. Also S_∞ is consistent by Proposition 2.4.6. We still need to show that S_∞ is maximal. Suppose $S \cup \{Z\}$ is consistent; we must show that $Z \in S_\infty$. Well, since $X_1, X_2, X_3, \dots$ is a listing of *all* formulas, Z is somewhere in the list; say Z is formula X_k . Then $S_k \cup \{X_k\}$ must be consistent, for if it were not, $S_\infty \cup \{X_k\}$ would not be consistent either, by Proposition 2.4.2, but this is $S_\infty \cup \{Z\}$ and our assumption is that this is consistent. Since $S_k \cup \{X_k\}$ is consistent, it is this set that constitutes S_{k+1} , hence $Z = X_k \in S_{k+1} \subseteq S_\infty$.

Exercises

Exercise 2.4.1 This really amounts to a test of your understanding of Definition 2.4.1. Suppose both $\mathcal{A}$ and $\mathcal{R}$ are empty, that is, we have no axioms and no rules of inference. Under these circumstances, what sets S would be consistent? And what would be the *maximally* consistent sets?

2.5 Implication and the Deduction Theorem

We now have all the necessary background, and have proved some important results. We have done this only using the general idea of an axiomatic proof system. We have, as yet, chosen no axioms or rules of inference. It is now time to start filling in the details of our axiomatization $\mathcal{A}$ of classical propositional logic.

Implication is often taken to be the most fundamental connective. We begin with an important result due independently to Tarski and Herbrand. It says that if we pick the ‘right’ axioms and rules for implication, we can prove what is called the *Deduction Theorem*, 2.5.1, telling us that derivation, from Definition 2.2.1, and implication are closely connected. This gives us the start of our axiom system. We begin with a derivation rule that is probably familiar to the reader.

Rule of Derivation, Modus Ponens

$$\frac{X \quad (X \supset Y)}{Y}$$

Next, we adopt two axiom schemes for the implication connective.

$$\supset -1 \quad (X \supset (Y \supset X))$$

$$\supset -2 \quad (X \supset (Y \supset Z)) \supset ((X \supset Y) \supset (X \supset Z))$$

From here on, $\mathcal{A}$ is an axiom system with at least this two schemes, and $\mathcal{R}$ is a set of rules that includes *Modus Ponens*. In fact, *Modus Ponens* is the only rule we will need for classical propositional logic. So from now on we stop mentioning $\mathcal{R}$, and concentrate on what our axiom schemes are. Let us give an (important) example of a proof using the axiom schemes $\mathcal{A}$ that we have so far. We will show $A \supset A$ is provable, for any formula A . We do not claim a proof is easy to discover, but here is such a proof. Line numbers are so that we can give explanations.

1. $A \supset ((W \supset A) \supset A)$
2. $(A \supset ((W \supset A) \supset A)) \supset ((A \supset (W \supset A)) \supset (A \supset A))$
3. $(A \supset (W \supset A)) \supset (A \supset A)$
4. $A \supset (W \supset A)$
5. $A \supset A$

In this line 1 is an axiom; in $\supset -1$ take X to be A and Y to be $W \supset A$, where W is any formula you may choose; it’s exact value won’t matter. Line 2 is an instance of $\supset -2$; take X to be A , Y to be $W \supset A$ and Z to be A . Line 3 follows from lines 1 and 2 by *Modus Ponens*. Line 4 is an instance of $\supset -1$, take X to be A and Y to be W . Finally, line 5 follows from lines 3 and 4 by *Modus Ponens*.

As we noted above, this proof is not easy to discover. In fact, $(A \supset B) \supset ((B \supset C) \supset (A \supset C))$ has a proof too, and you might convince yourselves that it is really difficult to find (bet you don’t find one, in fact). The power of the Deduction Theorem, given below, is that once we have it, we have easier ways of discovering such proofs. Recall that in Definition 2.2.1 we had a notion of derivation, as well as a notion of proof. It is the notion of derivation, for the axiom system $\mathcal{A}$ given so far, that plays a central role now.

Theorem 2.5.1 (Deduction Theorem) *Let S be a set of formulas, and let X and Y be single formulas. In axiom system $\mathcal{A}$ containing the axiom schemes given so*

far, and in every extension of it that results by adding further axiom schemes, if $S \cup \{X\} \vdash Y$, then $S \vdash (X \supset Y)$. Further, there is an algorithm for converting a derivation showing $S \cup \{X\} \vdash Y$ into one showing $S \vdash (X \supset Y)$.

Proof Let us assume we have a derivation of Y from $S \cup \{X\}$, say it looks like this:

$$\begin{array}{c} Z_1 \\ Z_2 \\ Z_3 \\ \vdots \\ Z_{n-1} \\ Z_n = Y \end{array}$$

In this derivation the last line is Y and otherwise, since it is a derivation from $S \cup \{X\}$, each line is either an axiom, or a member of $S \cup \{X\}$, or comes from earlier lines by *Modus Ponens*. Our goal is to turn this derivation into one of $X \supset Y$, that is, $X \supset Z_n$, but from S alone.

As a first step, append $X \supset$ to the beginning of each line, getting the following sequence.

$$\begin{array}{c} X \supset Z_1 \\ X \supset Z_2 \\ X \supset Z_3 \\ \vdots \\ X \supset Z_{n-1} \\ X \supset Z_n \text{ (that is, } X \supset Y) \end{array}$$

This has the last line we want, but it is very unlikely that it is a legal derivation. Now we add some lines to turn it into one.

Consider one of the lines above, say $X \supset Z_k$. We are going to insert some lines immediately before this. What lines depends on the justification for Z_k being in the original derivation, and so there are three different cases.

Case 1: Z_k is an axiom. Then replace the line $X \supset Z_k$ in the sequence by the following:

$$\begin{array}{c} Z_k \\ Z_k \supset (X \supset Z_k) \\ X \supset Z_k \end{array}$$

We can justify the presence of each of these. The first line is an axiom. The second line is an axiom, an instance of scheme $\supset - 1$. The third follows from the first two by *Modus Ponens*. All this is allowed in a derivation from S , indeed, in a derivation from anything.

Case 2: Z_k is a member of S . This is handled exactly as in Case 1. We expand the line in the same way. The only difference is that now the presence of Z_k is justified because it is a member of S , and so is allowed in a derivation from S .

Case 3: Z_k is X . In this case the formula $Z_k \supset X$ is just $X \supset X$. We know it has an axiomatic proof; we showed this earlier. Insert the steps of that proof just above the line $Z_k \supset X$.

Case 4: Z_k comes from earlier lines by *Modus Ponens*. That is, there are lines Z_i and Z_j where $i, j < k$, and $Z_j = (Z_i \supset Z_k)$. Then, in the list with $X \supset$ prefixed, we must have $X \supset Z_i$ and $X \supset (Z_i \supset Z_k)$ somewhere before the line $X \supset Z_k$. This time, expand the line $X \supset Z_k$ to the following.

$$\begin{aligned} (X \supset (Z_i \supset Z_k)) &\supset ((X \supset Z_i) \supset (X \supset Z_k)) \\ (X \supset Z_i) &\supset (X \supset Z_k) \\ X &\supset Z_k \end{aligned}$$

The first of these formulas is an axiom, an instance of scheme $\supset-2$. Since $X \supset (Z_i \supset Z_k)$ occurs somewhere earlier, the second line follows by *Modus Ponens*. And then, since $X \supset Z_i$ also occurs somewhere earlier, the third line also follows by *Modus Ponens*.

By adding these extra lines we have converted the sequence of formulas with $X \supset$ added into a proper derivation, and only members of S are now used as assumptions. (Note: very strictly speaking, the correctness of the conversion process given above is really an induction on proof length. In the interests of readability we have been somewhat relaxed in our presentation.)

Here is an example to show how useful this theorem can be. Earlier we said that $(A \supset B) \supset ((B \supset C) \supset (A \supset C))$ was provable, but it was hard to find a proof. Here's a way to find one. First, we show there is a *derivation* to justify the following.

$$\{A \supset B, B \supset C, A\} \vdash C$$

Here is the derivation, with line numbers added for convenience.

$$\begin{array}{l} 1 \ A \\ 2 \ A \supset B \\ 3 \ B \\ 4 \ B \supset C \\ 5 \ C \end{array}$$

In this, line 1 is one of the premises. So is line 2. Line 3 follows from 1 and 2 by *Modus Ponens*. Line 4 is a premise, and line 5 follows from 3 and 4 by *Modus Ponens*.

There are three uses of premises, each of which must be eliminated. We show how to handle line 1, and leave the rest to you. Following the algorithm given in the proof of Theorem 2.5.1, we first append $A \supset$ to each line, getting the following.

$A1 \ A \supset A$
 $A2 \ A \supset (A \supset B)$
 $A3 \ A \supset B$
 $A4 \ A \supset (B \supset C)$
 $A5 \ A \supset C$

Each of these lines needs formulas inserted above them, to justify their presence. We discuss two lines and leave the other three to you. The two lines we cover are $A1$ and $A3$. Then once the remaining lines are also taken care of we will have a legal derivation of $A \supset C$ from $\{A \supset B, B \supset C\}$. Line $A1$ is covered by case 3 and line $A3$ is covered by case 4 in the proof of the deduction theorem. Here is the result of treating these two lines.

$A \supset ((W \supset A) \supset A)$
 $(A \supset ((W \supset A) \supset A)) \supset ((A \supset (W \supset A)) \supset (A \supset A))$
 $(A \supset (W \supset A)) \supset (A \supset A)$
 $A \supset (W \supset A)$
 $A1 \ A \supset A$
 $A2 \ A \supset (A \supset B)$
 $(A \supset (A \supset B)) \supset ((A \supset A) \supset (A \supset B))$
 $(A \supset A) \supset (A \supset B)$
 $A3 \ A \supset B$
 $A4 \ A \supset (B \supset C)$
 $A5 \ A \supset C$

Once lines $A2$, $A4$ and $A5$ have been taken care of, we will have a properly constructed derivation showing the following

$$\{A \supset B, B \supset C\} \vdash A \supset C.$$

Next, another application of the algorithm from the proof of the Deduction Theorem will convert this derivation into one showing

$$\{A \supset B\} \vdash (B \supset C) \supset (A \supset C)$$

and then one more application of the algorithm will produce a derivation showing

$$\emptyset \vdash (A \supset B) \supset ((B \supset C) \supset (A \supset C))$$

and since this is a derivation from the empty set, we will have created a proof of $(A \supset B) \supset ((B \supset C) \supset (A \supset C))$, as promised. And, as promised, it is long and complex, which is why we haven't actually given the complete proof.

Remark We should note that our two axiom schemes and the rule of *Modus Ponens* don't quite give us everything needed for classical implication. In fact, what is axiomatized so far is the implication of *intuitionistic logic*. We don't prove that here, or even explain what it means. We're just notifying you.

Exercises

Exercise 2.5.1 Use the Deduction Theorem to establish the provability of $(A \supset (B \supset C)) \supset (B \supset (A \supset C))$. Then give an actual axiomatic proof.

2.6 The Other Connectives

So far our axiom system $\mathcal{A}$ has axiom schemas $\supset - 1$ and $\supset - 2$, and *Modus Ponens* as a rule of derivation. We now start the business of adding axioms for conjunction, disjunction, and negation. Recall, our goal is that maximally consistent sets should correspond to boolean valuations, where the correspondence is that being a member of a maximally consistent set, and mapping to **t** under a boolean valuation, should match up.

2.6.1 Conjunction

We must add enough axioms to $\mathcal{A}$ so that if M is a maximally consistent set then $(X \wedge Y) \in M$ if and only if $X \in M$ and $Y \in M$. Suppose we take as an axiom scheme $(X \wedge Y) \supset X$. Then if $X \wedge Y$ is in a maximally consistent set, X will also be present because maximally consistent sets are closed under derivation, Proposition 2.4.4. We can do a similar thing with the second component of a conjunction. So, we adopt the following two axiom schemes.

$$\wedge - 1 \quad ((X \wedge Y) \supset X)$$

$$\wedge - 2 \quad ((X \wedge Y) \supset Y)$$

Making sure that $X \in M$ and $Y \in M$ implies $(X \wedge Y) \in M$ is almost equally simple. We adopt the following axiom scheme.

$$\wedge - 3 \quad (X \supset (Y \supset (X \wedge Y)))$$

We now have the following.

Proposition 2.6.1 *Assume the three axioms for conjunction given above have been added to the axiom system $\mathcal{A}$. Then, if M is any maximally consistent set, $(X \wedge Y) \in M$ if and only if $X \in M$ and $Y \in M$.*

Note that the axiom schemas $\wedge-1$, $\wedge-2$, and $\wedge-3$ just added to $\mathcal{A}$ are tautologies, so $\mathcal{A}$ as constructed so far is sound.

2.6.2 Disjunction

Disjunction is more complicated than conjunction. We want axioms to ensure that a maximally consistent set M meets the condition: $(X \vee Y) \in M$ if and only if $X \in M$ or $Y \in M$. Actually, one direction is simple; the one from right to left. We adopt the following axiom schemes (which are tautologies).

$$\vee-1 \quad (X \supset (X \vee Y))$$

$$\vee-2 \quad (Y \supset (X \vee Y))$$

We still need the other direction to hold: if $(X \vee Y) \in M$ then $X \in M$ or $Y \in M$. Let us look at it in the contrapositive direction: if $X \notin M$ and $Y \notin M$ then $(X \vee Y) \notin M$. Suppose we try proving this, and see what needs to be added to make the proof go through.

Assume $X \notin M$ and $Y \notin M$. We want to reason our way to $(X \vee Y) \notin M$. Since $X \notin M$ and M is *maximally* consistent, $M \cup \{X\}$ must be inconsistent. Then there must be some finite subset of M , call it S_X , such that $S_X \cup \{X\} \vdash Z$ for every formula Z . Similarly since $Y \notin M$ there must be some finite subset of M , call it S_Y , such that $S_Y \cup \{Y\} \vdash Z$ for every formula Z . Of course $S_X \cup S_Y$ is also a finite subset of S and, using Proposition 2.2.2, both $S_X \cup S_Y \cup \{X\} \vdash Z$ and $S_X \cup S_Y \cup \{Y\} \vdash Z$ hold for every formula Z .

Using the Deduction Theorem, 2.5.1, $S_X \cup S_Y \vdash (X \supset Z)$ and $S_X \cup S_Y \vdash (Y \supset Z)$, for every formula Z . If we could somehow get from these to $S_X \cup S_Y \vdash ((X \vee Y) \supset Z)$, then we could conclude $S_X \cup S_Y \cup \{X \vee Y\} \vdash Z$ for every Z . And since $S_X \cup S_Y$ is a finite subset of M , this would tell us that $M \cup \{X \vee Y\}$ is inconsistent, and hence that $(X \vee Y) \notin M$, which is what we are after. The missing part of the argument, labeled, is a way of getting from $(X \supset Z)$ and $(Y \supset Z)$ to $((X \vee Y) \supset Z)$. Well, why don't we simply adopt the following axiom scheme which, helpfully, is a tautology.

$$\vee-3 \quad ((X \supset Z) \supset ((Y \supset Z) \supset ((X \vee Y) \supset Z)))$$

Proposition 2.6.2 *Assume the axioms $\vee-1$, $\vee-2$ and $\vee-3$, for disjunction have been added to the axiom system. Then if M is any maximally consistent set, $(X \vee Y) \in M$ if and only if $X \in M$ or $Y \in M$.*

From now on $\vee-1$, $\vee-2$, and $\vee-3$ are taken to be part of axiom system $\mathcal{A}$.

2.6.3 Negation

A boolean valuation maps exactly one of X or $\neg X$ to **t**. Corresponding to this, we would like maximally consistent sets of formulas to contain exactly one of X and $\neg X$ for each formula X . The axioms we now add to $\mathcal{A}$ handle this quite directly. The first item says that we can't have both X and $\neg X$ because if we did we would have anything, hence inconsistency. The second says we must have one of X or $\neg X$. All this is informal, but a formal proof follows the statement of the axioms. From now on axiom system $\mathcal{A}$ includes the following.

$$\neg-1 \quad (\neg X \supset (X \supset Y))$$

$$\neg-2 \quad (X \vee \neg X)$$

Proposition 2.6.3 *Let M be a set of formulas that is maximally consistent set using axiom system $\mathcal{A}$. For each formula X , exactly one of X or $\neg X$ is in M .*

Proof First, suppose we had both X and $\neg X$ in the maximally consistent set M . Then $\{X, \neg X\}$ would be a finite subset of M and, using $\neg-1$ and *Modus Ponens*, $\{X, \neg X\} \vdash Y$, for every formula Y , so M would not be consistent. So, we can't have both present.

Second, for any formula X , $X \vee \neg X$ is now an axiom, so $(X \vee \neg X) \in M$ by Corollary 2.4.5. Then $X \in M$ or $\neg X \in M$ by Proposition 2.6.2.

2.6.4 Implication

We have completed our axiom system $\mathcal{A}$, but it still remains for us to show the implication connective has the properties we need. A boolean valuation maps $X \supset Y$ to **t** exactly when it either does not map X to **t** or does map Y to **t**. If maximally consistent sets are to correspond to boolean valuations we should have a counterpart of this.

Proposition 2.6.4 *Let M be a maximally consistent set of formulas using axiom system $\mathcal{A}$. Then $(X \supset Y) \in M$ if and only if $X \notin M$ or $Y \in M$.*

Proof Assume M is maximally consistent.

Left to right. Suppose $(X \supset Y) \in M$. If $X \notin M$ we are done. If $X \in M$ then it follows that $M \vdash Y$, and hence $Y \in M$ because M is *maximally* consistent and we have Proposition 2.4.4.

Right to left. If $Y \in M$ then $(X \supset Y) \in M$ because maximally consistent sets are closed under derivability, and $Y \supset (X \supset Y)$ is an axiom, $\supset-1$. If $X \notin M$ then by Proposition 2.6.3, $\neg X \in M$. Then much like before, $(X \supset Y) \in M$ because $\neg X \supset (X \supset Y)$ is an axiom $\neg-1$.

Exercises

Exercise 2.6.1 Give an axiomatic proof of $(A \wedge B) \supset (B \wedge A)$.

Exercise 2.6.2 Supply a proof for Proposition 2.6.1.

Exercise 2.6.3 Give an axiomatic proof of $(A \vee B) \supset (B \vee A)$.

Exercise 2.6.4 Give a proper proof that, using axiom system $\mathcal{A}$, for a maximally consistent set M , if $X \in M$ or $Y \in M$ then $(X \vee Y) \in M$.

Exercise 2.6.5 Give an axiomatic proof in $\mathcal{A}$ of $(X \supset (\neg X \supset Y))$. Hint: use the deduction theorem.

Exercise 2.6.6 Give an axiomatic proof of $\neg\neg X \supset X$. Hint: use $\vee-3$ and the deduction theorem.

Exercise 2.6.7 Give an axiomatic proof of $(\neg A \vee B) \supset (A \supset B)$.

Exercise 2.6.8 Give an axiomatic proof of $(A \supset B) \supset (\neg A \vee B)$.

2.7 Summary of Our Classical Axiom System

Axiom system $\mathcal{A}$ has been presented over several sections. Here is the whole system in one place.

Rule of Derivation: *Modus Ponens*

$$\frac{X \quad X \supset Y}{Y}$$

Axioms:

Name	Axiom Scheme
$\supset-1$	$(X \supset (Y \supset X))$
$\supset-2$	$((X \supset (Y \supset Z)) \supset ((X \supset Y) \supset (X \supset Z)))$
$\wedge-1$	$((X \wedge Y) \supset X)$
$\wedge-2$	$((X \wedge Y) \supset Y)$
$\wedge-3$	$(X \supset (Y \supset (X \wedge Y)))$
$\vee-1$	$(X \supset (X \vee Y))$
$\vee-2$	$(Y \supset (X \vee Y))$
$\vee-3$	$((X \supset Z) \supset ((Y \supset Z) \supset ((X \vee Y) \supset Z)))$
$\neg-1$	$(\neg X \supset (X \supset Y))$
$\neg-2$	$(X \vee \neg X)$

Exercises

Exercise 2.7.1 We did not consider equivalence, $\equiv$, as a basic connective, but we could have. We could also have considered the Sheffer stroke connective (also known as NAND), $\uparrow$, or the joint denial connective (also known as NOR), $\downarrow$. Here is how they behave semantically—that is, here are the boolean valuation conditions for them.

	$\equiv$	$\uparrow$	$\downarrow$
t t	t	f	f
t f	f	t	f
f t	f	t	f
f f	t	t	t

1. Invent appropriate axiom schemes for $\equiv$ and show the completeness proof extends to incorporate them. That is, one can prove completeness where formulas are allowed to contain $\equiv$ as a connective.
2. Do the same for $\uparrow$.
3. Do the same for $\downarrow$.

2.8 Completeness At Last

We have now finished putting together our axiom system $\mathcal{A}$. Every one of our axioms is a tautology, and *Modus Ponens* produces tautologies from tautologies, so the system is sound by Proposition 2.3.2. We have carefully chosen our axioms so that we have what we need to prove completeness. All that remains is to actually give, in a coherent form, the proof we have been hinting at.

Assume formula X does not have a proof. Suppose we could find a maximally consistent set M that did not contain X . Then we could easily show X was not a tautology as follows. Define a mapping v by setting $v(Z) = \mathbf{t}$ if $Z \in M$ and $v(Z) = \mathbf{f}$ if $Z \notin M$, for every formula Z . We have adopted axiom schemes that guarantee v will be a boolean valuation. Then since $X \notin M$ it must be that $v(X) = \mathbf{f}$, and so X is not a tautology. So it is enough to establish that if X does not have a proof, there is some maximally consistent set M that omits X .

By Proposition 2.6.3, M will omit X just in case M contains $\neg X$. So we must find a maximally consistent set containing $\neg X$. Lindenbaum's Lemma, 2.4.7, provides us with a way of creating maximally consistent sets. According to it, if $\{\neg X\}$ were consistent, it could be extended to a maximally consistent set that does not contain X , and we would be done. So all that is left is, we must establish that if X is not provable, then $\{\neg X\}$ is consistent. What we will actually show is the contrapositive: if $\{\neg X\}$ is not consistent then X is provable.

Assume $\{\neg X\}$ is not consistent. Well, to be inconsistent there must be some finite subset of $\{\neg X\}$ that has all formulas as consequences. There are only two subsets, the empty set, $\emptyset$, and the entire set, $\{\neg X\}$. We treat them separately.

If the empty set has all formulas as consequences, of course X would be a consequence. But the consequences of the empty set are just the theorems. So X would be a theorem in this case.

Now suppose it is $\{\neg X\}$ that has all formulas as consequences. Then, in particular, it must have X as a consequence, that is, $\{\neg X\} \vdash X$. Using the Deduction Theorem, 2.5.1, $\neg X \supset X$ is then provable. Now the formula $(X \supset X) \supset ((\neg X \supset X) \supset ((X \vee \neg X) \supset X))$ is an instance of Axiom Scheme $\vee-3$. We proved that $X \supset X$ is a theorem, so by *Modus Ponens*, $(\neg X \supset X) \supset ((X \vee \neg X) \supset X)$ is a theorem. Since $\neg X \supset X$ is provable, by *Modus Ponens* again, $(X \vee \neg X) \supset X$ is a theorem. Finally, we have $X \vee \neg X$ by Axiom Scheme $\neg-1$, and by using *Modus Ponens* one last time, we have that X is a theorem.

So to recapitulate, if X is not provable then $\{\neg X\}$ is consistent. It extends to a maximally consistent set M which cannot contain X . M can be used to create a corresponding boolean valuation, and since X is not present, X maps to **f**, and thus is not a tautology.

Theorem 2.8.1 (Axiomatic Completeness) *Assume axiom system $\mathcal{A}$ has all the axiom schemes given so far, and the Modus Ponens rule. If X is a tautology, then X has an axiomatic proof.*

2.9 Redefining Consistency

When we defined inconsistency and consistency in an axiomatic setting, Definition 2.4.1, it was somewhat convoluted. Recall, a set was inconsistent if it had a finite subset with every formula as a consequence. Why didn't we simply say a set was inconsistent if one could derive a contradiction from it? The reasons for this are pedagogical. We wanted the desired properties of maximally consistent sets to serve us as a guide in our choice of axioms. At the beginning we had chosen no axioms at all, and so we had no machinery to characterize what a contradiction might be. So we chose a definition of consistency that was extremely general and made sense whatever axioms and connectives we eventually might settle on. But now everything is in place. We can say what constitutes a contradiction (we use $P \wedge \neg P$), and show that a definition making use of it is equivalent to the one we used *in the presence of the axioms we actually wound up choosing*. This allows us to replace inconsistency as implying everything with inconsistency as implying the single formula $P \wedge \neg P$, often a simpler thing to manage.

Lemma 2.9.1 *Using the axiom system from Sect. 2.7, any formula of the form $X \wedge \neg X$ implies every formula.*

Proof Let $A \wedge \neg A$ be a formula of this form. We show $\{A \wedge \neg A\} \vdash Z$ for any Z and so $(A \wedge \neg A) \supset Z$ is a theorem of our axiom system, by the Deduction Theorem. Here is the simple derivation.

$(A \wedge \neg A)$	assumption
$(A \wedge \neg A) \supset A$	$\wedge - 1$
A	<i>Modus Ponens</i>
$(A \wedge \neg A) \supset \neg A$	$\wedge - 2$
$\neg A$	<i>Modus Ponens</i>
$\neg A \supset (A \supset Z)$	$\neg - 1$
$A \supset Z$	<i>Modus Ponens</i>
Z	<i>Modus Ponens</i>

Note that since Z could be $B \wedge \neg B$, it follows that any two formulas of the form $X \wedge \neg X$ can be proved to be equivalent. Consequently it doesn't really matter which formula of this form we choose to work with. Now we have an equivalent for our earlier definition of consistency that may strike you as more intuitive.

Proposition 2.9.2 *Using the axiom system of Sect. 2.7 a set S of formulas is inconsistent as in Definition 2.4.1 if and only if $S \vdash (P \wedge \neg P)$.*

Proof If S is inconsistent as in Definition 2.4.1, S has a finite subset S_0 that has every formula as a consequence, in particular, $S_0 \vdash (P \wedge \neg P)$. But then $S \vdash (P \wedge \neg P)$ using Proposition 2.2.2.

If $S \vdash (P \wedge \neg P)$ there is some finite subset $S_0 \subseteq S$ such that $S_0 \vdash (P \wedge \neg P)$ by Proposition 2.2.3. Then $S_0 \vdash X$ for every formula X by the Lemma above, so S is inconsistent, as in Definition 2.4.1.

Chapter 3

Background: Propositional Tableaus



The semantics for classical propositional logic is the simplest of any active logic—models are specified using truth tables, or equivalently boolean valuations. In Chap. 2 an axiomatization was developed. Semantics and axiomatic proofs fit tightly together—we have soundness and completeness theorems. But axiom systems are hard to use—discovering a proof can be difficult. Indeed, the axioms needed for a particular proof might be much more complicated than the formula being proved. But there are many kinds of proof procedures. Those that are sometimes called *analytic* have the very nice property that the formulas that go into a proof all are parts of the formula being proved (or perhaps also negations of such formulas, as is the case here). We are about to present an analytic proof system called *tableaus* for classical propositional logic. (Tableaus are also often known as *tree proof systems*.) In much of the rest of this book tableaus will be our primary proof systems. A tableau proof proceeds by decomposing the formula being proved into simpler and simpler parts, and thus a proof construction is generally easier since our choices of what to do next are limited. Also they provide us with explicit counter models for unprovable formulas, and do so quite directly, while our axiomatic completeness proof merely establishes the existence of a counter model for an unprovable formula, without really telling us what it is. Finally, motivation for the rules of a tableau proof system often can be extracted quite directly from the intended semantics.

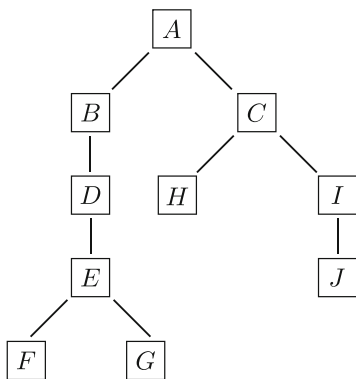
The classical reference for classical tableaus is Smullyan (1968). One of the authors of the book you are reading has some fondness for Fitting (1996), but general treatments can be found in many places, such as Jeffrey (1967), Nerode and Shore (1997), Howson (1997), Smullyan (2009), Smullyan (2014), and Gasquet et al. (2014). There is more than you probably want to know in D’Agostino et al. (1999), and perhaps about the right amount in Wikipedia’s article on *analytic tableaus*. Tableau systems actually come in several versions—see Fitting (1999) for some discussion of their history and variety. Just as with axiom systems, tableaus are on their best behavior in the case of classical propositional logic, so it is a good place to begin.

3.1 Tableaus

Tableau systems are *refutation systems*. To prove a formula we begin by assuming the opposite, analyze the consequences using a kind of tree structure and, if the consequences turn out to be impossible, we conclude the opposite of our original formula can never hold, and so the formula itself must be valid. Of course this is very informal, and will be replaced by a proper formulation later on.

Some treatments of tableaus use *signed* formulas, TX and FX where X is a formula, where these are read informally as asserting truth, or falsity, of X . For intuitionistic and some other non-classical logics, signs have a different intuition, and play an essential role, but for classical logic they can be dispensed with. We can simply drop T , and replace F with negation, and we have chosen to do this here.

A tableau proof has a *tree* structure. Technically, a *tree* is a kind of graph. Rather than giving a formal definition, we give a representative example which should suffice to convey the general idea.



The lettered square boxes are called *nodes* and the lines are *edges* which should be thought of as directed downward. In this chapter nodes will be labeled with classical propositional formulas. In the diagram here, the node (labeled) A is the *root node* and nodes F , G , H , and J are *leaves*. (Thus botanically, our trees are upsidedown.) Node A has two *children*, B and C , with B the *left child* and C the *right child*. Nodes C and E also have two children; all other nodes that are not leaves have just one child each. (Our nodes will almost always have at most two children, though towards the end of the book nodes with three children will appear.) If one node is the child of another, the second is the *parent* of the first—for example, C is the parent of both H and I . The sequence A, B, D, E is an example of a *path*. The sequence A, B, D, E, F is another path and it is also a *branch*, which means it is a maximal path. Trees can be infinite, but such trees will not come up for some time, so we can ignore them for now.

An attempt to construct a tableau proof of a formula Z of classical propositional logic begins by creating a tree with $\neg Z$ at its root (and with no other nodes). Intuitively, the formula at the root represents the possibility that Z is false (that is,

$\neg Z$ is true) under some boolean valuation. Next, branches are “grown” according to certain *Branch Extension Rules*, to be given shortly. This yields a succession of *tableaus* for $\neg Z$. It may happen that each branch of some resulting tableau is *closed*, something we define properly after we give the rules for growing the branches. If each branch is closed, this represents an impossible situation, so we conclude the original assumption that Z could be false under some boolean valuation cannot be the case. Then Z must be a tautology. Of course all this is an informal description, and we continue informally for a bit longer before we move to the proper formal version.

Suppose we have $X \wedge Y$ at a tableau node. Intuitively this says $X \wedge Y$ is true under the boolean valuation we (informally) started with. Then both X and Y are also true, and so we should allow ourselves to add X and Y to the branch. That is, any branch going through a node with $X \wedge Y$ can be lengthened by adding a new node with X to the end, then another after that, with Y . This is one example of a Branch Extension Rule. If we have a node with $X \vee Y$, intuitively $X \vee Y$ is true under our boolean valuation, so at least one of X or Y is also true, and so we may also have one of X or Y on the branch. We accomplish this by giving a branch going through the node with $X \vee Y$ *two* children, a node with X as left child and a node with Y as right child. This is another example of a Branch Extension Rule.

All the binary propositional connectives we use fit one or the other of these patterns. We group them into conjunctive cases (like $X \wedge Y$), disjunctive cases (like $X \vee Y$), and a (double) negation case. The vertical line in the conclusions of the disjunctive cases represents tableau branching.

Definition 3.1.1 (Classical Propositional Branch Extension Rules)

$\frac{X \wedge Y}{X}$	$\frac{\neg(X \vee Y)}{\neg X}$	$\frac{\neg(X \supset Y)}{X}$	$\frac{X \vee Y}{X \mid Y}$	$\frac{\neg(X \wedge Y)}{\neg X \mid \neg Y}$	$\frac{X \supset Y}{\neg X \mid Y}$	$\frac{\neg\neg X}{X}$
Y	$\neg Y$	$\neg Y$				
Conjunctive Rules			Disjunctive Rules		Negation Rule	

It is important to point out that the branch extension rules are *non-deterministic*. A rule can be applied to any formula on a branch that matches one of the patterns above the lines in Definition 3.1.1. Formulas that are added always go at branch ends. Now finally, how do tableau proofs start and finish?

Definition 3.1.2 For classical propositional tableaus:

Closed Branch A tableau branch is *closed* if it contains both A and $\neg A$ for some formula A . If A is atomic, the branch is called *atomically* closed. A branch that is not closed is *open*.

Closed Tableau A tableau is (*atomically*) *closed* if every branch is (atomically) closed.

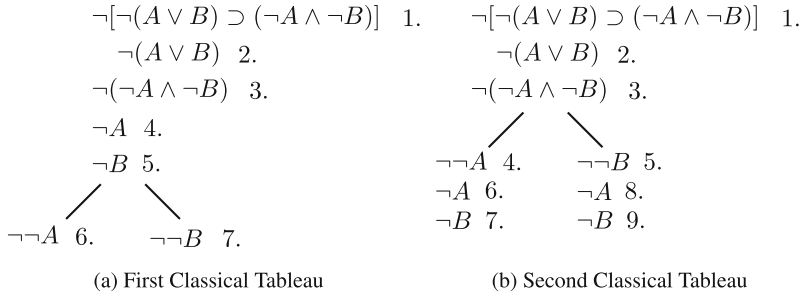


Fig. 3.1 Two tableau proofs of the same formula

Tableau Proof A closed tableau for $\neg Z$ (that is, beginning with $\neg Z$) is a *proof* of Z , and Z is a *theorem*.

A closed branch represents an impossible state of affairs—there is no boolean valuation in which all the formulas on it could be true. A closed tableau for $\neg Z$, in effect, says that no branch represents a situation that is possible. All this is assuming Z could be false under some circumstances, since we began the tableau with $\neg Z$. A closed tableau tells us such an assumption leads to a contradiction.

Note that there is a difference between closure and atomic closure, which is a stricter requirement. In fact, it will be shown that there is a closed tableau proving a formula Z if and only if there is an atomically closed formula proving Z . So in trying to show provability, finding a closed tableau is enough. For some purposes the additional information an atomically closed tableau contains is needed.

Example 3.1.3 Figure 3.1 shows two tableau proofs of the same formula, $\neg(A \vee B) \supset (\neg A \wedge \neg B)$, which is one of De Morgan’s laws. We begin with the negation of the formula, then apply tableau rules. In this display we do what we will continue to do throughout the book. Initial negations that are present because of a tableau condition, but did not appear in the original formula, are ‘outdented’. In the present example, this is all initial occurrences of negation except one. We have numbered tableau nodes for convenience in talking about them.

In both Fig. 3.1a, b we begin with the negation of $\neg(A \vee B) \supset (\neg A \wedge \neg B)$, line 1. In both versions, 2 and 3 are from 1 by the rule for negated implication, Rule $\neg \supset$. (We use this sort of shorthand notation for rule names throughout the book.) Then in 3.1a, lines 4 and 5 are from line 2 by Rule $\neg \vee$; 6 and 7 are from 3 by Rule $\neg \wedge$. In 3.1b, 4 and 5 are from 3 by Rule $\neg \wedge$, and 6, 7 and also 8, 9 are from 2 by Rule $\neg \vee$.

In both tableaus of Fig. 3.1, the left branches are closed because they contain $\neg A$ and its negation, $\neg\neg A$. The right branches are similarly closed, via $\neg B$ and $\neg\neg B$. No branch is *atomically* closed, but it is easy to see how to continue each tableau to atomic closure.

Just as axiom systems do not provide *unique* proofs for theorems, the same is true for tableaus as well. The two proofs in Fig. 3.1 illustrate this. As theorems get more complicated, the range of possible proofs grows significantly, and some are simpler than others. This is an issue in automated theorem proving, but is not of concern here.

Exercises

Exercise 3.1.1 Give at least two tableau proofs of $((A \wedge \neg B) \supset C) \supset (\neg C \supset (A \supset B))$.

Exercise 3.1.2 Of the two tableaus in Fig. 3.1, one is shorter than the other. Suggest a strategy for the order in which rules are applied that will favor construction of shorter proofs over longer ones.

Exercise 3.1.3 In both tableaus of Fig. 3.1, branches are closed because they contain a formula and its negation, *where the formula is not atomic*. Prove that *every* closed tableau can be continued to one that is atomically closed.

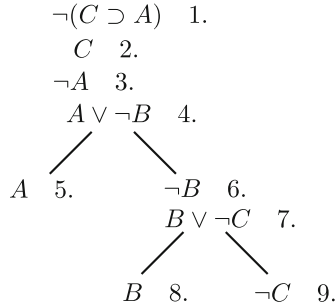
3.2 Logical Consequence Using Tableaus

In Definition 2.2.1 we introduced logical consequence for axiomatically formulated classical logic, $S \vdash X$, where S was a set of formulas and X was a single formula. Loosely, the idea was that members of S , which are particular formulas and not schemes, are treated like additional axioms. The corresponding version for tableaus is equally simple.

Definition 3.2.1 (Tableau Consequence) Let S be a set of formulas and X be a single formula. We say X is a *consequence* of S using tableaus, also written $S \vdash X$, if there is a closed tableau for $\neg X$ allowing the additional rule: If Y is any member of S then Y can be added to the end of any open branch. We call such a tableau a *derivation from S* , rather than a proof.

Loosely, the tableau rule above just amounts to allowing ourselves to assume the truth of any member of S , as needed. Here is an example of a tableau derivation.

Example 3.2.2 We show using tableaus that $\{A \vee \neg B, B \vee \neg C\} \vdash C \supset A$, that is, $S \vdash X$ where S is the set whose members are $A \vee \neg B$ and $B \vee \neg C$ and X is the formula $C \supset A$.



In this, 4 and 7 are introduced as members of the set of assumptions, and 1 is the negation of the formula we are deriving. The rest follow by various tableau rules.

Exercises

Exercise 3.2.1 Show, using tableaus, that $\{A \supset (B \vee C), C \supset B, \neg B\} \vdash \neg A$.

Exercise 3.2.2 Verify that Propositions 2.2.2 and 2.2.3 hold for tableau consequence, just as they did for the axiomatic version.

3.3 Tableau Soundness

Soundness for a proof procedure means the procedure cannot prove any formula it should not. Then to say the classical tableau rules are sound means: if a formula X has a proof using the classical propositional tableau rules, then X must be a tautology. The soundness proof given here is the prototype for tableau soundness proofs in general, throughout this book.

Definition 3.3.1 (Satisfiable) A set S of classical propositional formulas is *satisfiable* if there is a boolean valuation v such that $v(X) = \mathbf{t}$ for every $X \in S$. A tableau branch is satisfiable if the set of formulas on it is satisfiable. And a tableau is satisfiable if some branch of it is satisfiable.

Very informally, a set of formulas is satisfiable if there is some way they could all be true at once. A tableau branch is treated like the conjunction of the formulas on it, and a tableau itself like the disjunction of its branches. We need two fundamental results about satisfiability, then soundness follows easily. The first is quite simple.

Proposition 3.3.2 *A closed tableau is not satisfiable.*

Proof Suppose we had a tableau that was both closed and satisfiable. Since it is satisfiable, some branch of it is. Let S be the set of formulas on that branch, and say all members of S map to $\mathbf{t}$ using the boolean valuation v . Since the tableau is also

closed, for some formula X both X and $\neg X$ are on the branch, and hence in S . But then both $v(X) = \mathbf{t}$ and $v(\neg X) = \mathbf{t}$. But also $v(\neg X) = \neg v(X) = \neg \mathbf{t} = \mathbf{f}$, and this is not possible.

The second result needs a little more work to prove.

Proposition 3.3.3 *If a tableau branch extension rule is applied to a satisfiable tableau, the result is another satisfiable tableau.*

Proof Assume $\mathcal{T}$ is a satisfiable tableau, and a branch extension rule (see Definition 3.1.1) is applied to it. More specifically, say a branch extension rule is applied to a formula on a branch $\mathcal{B}$ of tableau $\mathcal{T}$. The proof divides into several cases.

We begin with a case that is essentially trivial. Some branch of $\mathcal{T}$ is satisfiable. If a branch $\mathcal{B}'$ that is different from $\mathcal{B}$ is satisfiable, applying a branch extension rule on $\mathcal{B}$ will not affect $\mathcal{B}'$, so after the rule application we still have a satisfiable branch, $\mathcal{B}'$, hence a satisfiable tableau.

For the rest of the proof, we can assume that $\mathcal{B}$ itself is satisfiable. Say the members of the set of formulas on $\mathcal{B}$ are all mapped to $\mathbf{t}$ by the boolean valuation v . Now we have a case for each of the tableau rules that we may apply.

Conjunctive Cases Suppose $X \wedge Y$ occurs on $\mathcal{B}$, and we add X and Y to the end of $\mathcal{B}$. Since $\mathcal{B}$ is satisfiable using boolean valuation v , we have $v(X \wedge Y) = \mathbf{t}$. But then by definition of boolean valuation, $v(X) = \mathbf{t}$ and $v(Y) = \mathbf{t}$. It follows that $\mathcal{B}$, with X and Y added, is satisfiable using the same boolean valuation v . (All the other conjunctive cases are treated similarly.)

Disjunctive Cases Suppose $X \vee Y$ occurs on $\mathcal{B}$, and we split the end of the branch, adding X to the left fork and Y to the right. Because $\mathcal{B}$ was satisfiable using v , we have $v(X \vee Y) = \mathbf{t}$. Then by the behavior of boolean valuations, we have at least one of $v(X) = \mathbf{t}$ or $v(Y) = \mathbf{t}$. In the first case, the left extension of $\mathcal{B}$ is satisfiable. In the second case, the right extension is satisfiable. Either way, one of the branch extensions of $\mathcal{B}$ is satisfiable, so some branch of the extended tableau is satisfiable, and so the tableau itself is satisfiable.

(The other disjunctive cases are similar.)

Negation Case This is straightforward, and we omit discussion.

Now soundness of the tableau rules follows easily.

Theorem 3.3.4 (Tableau Soundness) *If X has a tableau proof (not necessarily atomically closed) using the classical propositional rules, X is a tautology.*

Proof We show the contrapositive: if X is not a tautology, then X has no tableau proof.

Assume X is not a tautology, and so there is a boolean valuation v so that $v(X) = \mathbf{f}$. Let us attempt to construct a tableau proof of X . We begin with a single branch containing only $\neg X$. Since we have $v(\neg X) = \neg v(X) = \neg \mathbf{f} = \mathbf{t}$, we are beginning our tableau construction with a satisfiable tableau. By Proposition 3.3.3, the result of applying any tableau rule to a satisfiable tableau is another satisfiable tableau, so every subsequent tableau in our proof attempt must be satisfiable. Then

by Proposition 3.3.2, no subsequent tableau can be closed. Since there can be no closed tableau starting with $\neg X$, X has no tableau proof.

Exercises

Exercise 3.3.1 Extend the work of this section to show that if $S \vdash X$ using classical propositional tableaus (Definition 3.2.1) then $S \models X$ (Definition 2.1.3).

3.4 Tableau Completeness

If a tableau system is sound, it doesn't prove anything it shouldn't. We also need that it proves everything it should. Specifically, we want to show that if X is a tautology then X has a classical propositional tableau proof. In fact, we will actually prove the stronger fact that if X is a tautology, then X has an atomically closed tableau proof. Our proof, like most completeness arguments, proceeds in the contrapositive direction, and we will show that if X has no atomically closed tableau proof, there is a boolean valuation that maps X to **f**, so X is not a tautology. We will present two different ways of showing this. One gives us a decision procedure for classical propositional logic. The other does not, but it is much like the completeness argument we gave for the classical axiom system of Chap. 2, and is easier to use when we come to more complex modal rules later on. We begin with material concerning what are sometimes called Hintikka sets, material that both proofs have in common. Then we move to things on which the proofs differ.

3.4.1 Hintikka Sets

In the completeness proof from Chap. 2, for an axiom system, we showed that a maximally consistent set of formulas is satisfiable, where consistency was defined axiomatically. One can abstract from this a very nice and purely structural condition that suffices—the full strength of maximal consistency can be weakened. This approach was developed by Hintikka, hence the name Hintikka set. Hintikka sets are also called *downward saturated* sets, for reasons that will be obvious.

Definition 3.4.1 (Hintikka Set) A set H of propositional formulas is a *Hintikka set* if it meets the following conditions.

1. H does not contain both A and $\neg A$ for any atomic formula A .
2. H meets the following closure conditions:

$$\begin{aligned}
X \wedge Y \in H &\implies X \in H \text{ and } Y \in H \\
\neg(X \wedge Y) \in H &\implies \neg X \in H \text{ or } \neg Y \in H \\
X \vee Y \in H &\implies X \in H \text{ or } Y \in H \\
\neg(X \vee Y) \in H &\implies \neg X \in H \text{ and } \neg Y \in H \\
X \supset Y \in H &\implies \neg X \in H \text{ or } Y \in H \\
\neg(X \supset Y) \in H &\implies X \in H \text{ and } \neg Y \in H \\
\neg\neg X \in H &\implies X \in H
\end{aligned}$$

Condition 1 is a consistency condition at the atomic level. The closure conditions of part 2 all say that if certain formulas are present, so are certain simpler formulas. The connection with tableau rules should be obvious.

The maximally consistent sets of Chap. 2 are all Hintikka sets. But maximally consistent sets will all be infinite, because they all meet the condition that if X is not present, $\neg X$ must be. Hintikka sets can be finite. As an extreme example, the empty set is a Hintikka set. So is the following, where A and B are atomic.

$$\{\neg(A \supset \neg(B \vee C)), A, \neg\neg(B \vee C), B \vee C, C\}$$

Of course Hintikka sets can be infinite too. The set of all propositional letters is a simple example of one.

Proposition 3.4.2 (Hintikka's Lemma) *Every Hintikka set is satisfiable.*

Proof Let H be a Hintikka set. We noted in Chap. 2 that a boolean valuation is completely specified by its behavior at the atomic level, and we make use of this now. Define a boolean valuation v as follows. For each propositional letter P , if $P \in H$ then let $v(P) = \mathbf{t}$, and otherwise let $v(P) = \mathbf{f}$. This fully determines v . Now we show that for each formula X ,

$$\begin{aligned}
X \in H &\text{ implies } v(X) = \mathbf{t} \\
\neg X \in H &\text{ implies } v(X) = \mathbf{f}
\end{aligned}
\tag{\star}$$

We need both parts of $(\star)$ to make the proof go through, but it is the first part that tells us H is satisfiable, by v .

To prove that all formulas satisfy $(\star)$ we use Complete Induction on formula degree. (Complete Induction is Proposition 1.3.1, and formula degree is Definition 1.3.2). Let us say the number n has property $\mathcal{P}$ if, for every formula X of degree n , X meets both the conditions of $(\star)$. We show that every number has property $\mathcal{P}$, and hence that every formula makes conditions $(\star)$ true.

Assume as our induction hypothesis that the conditions in $(\star)$ are known to hold for formulas with degree less than n , that is, numbers smaller than n have property $\mathcal{P}$. Let X be an arbitrary formula of degree n . We show conditions $(\star)$ hold for X too and, since X was arbitrary, it will follow that n has property $\mathcal{P}$ too. Then by Complete Induction it will follow that every number has property $\mathcal{P}$, and so every formula satisfies the conditions $(\star)$.

So, assume X is a formula of degree n , and conditions $(\star)$ hold for formulas of lower degree. The proof now divides into several cases, depending on the form of X . In what follows we refer to the first part of $(\star)$ as the *positive* condition, and the second as the *negative* one.

Atomic Case Suppose X is atomic, say it is the propositional letter P . For this case there are no formulas of lower degree, but we can prove what we want directly. The positive condition is the case by the definition of v . For the negative condition, if $\neg P \in H$, then $P \notin H$ by condition 1 of the definition of Hintikka set. so then $v(P) = \mathbf{f}$, again directly by the definition of v .

Negation Case Suppose $X = \neg A$, where we assume that A , being of lower degree than X , meets the conditions $(\star)$. For the positive condition for X , if $X \in H$ then $\neg A \in H$, so $v(A) = \mathbf{f}$ since A meets the negative condition of $(\star)$, but then $v(X) = v(\neg A) = \neg v(A) = \neg \mathbf{f} = \mathbf{t}$. For the negative condition for X , if $\neg X \in H$ then $\neg\neg A \in H$, and since H is a Hintikka set, $A \in H$. Since A meets $(\star)$, $v(A) = \mathbf{t}$, so $v(X) = v(\neg A) = \neg v(A) = \neg \mathbf{t} = \mathbf{f}$.

BINARY OPERATION CASE We only show the result for implication; conjunction and disjunction are similar. Suppose $X = A \supset B$ and both A and B , being of lower degree, meet the conditions $(\star)$. For the positive condition for X , if $X \in H$ then $A \supset B \in H$, which is a Hintikka set, so $\neg A \in H$ or $B \in H$. Since A and B meet $(\star)$, $v(A) = \mathbf{f}$ or $v(B) = \mathbf{t}$. Either way, $v(A \supset B) = v(A) \supset v(B) = \mathbf{t}$, so $v(X) = \mathbf{t}$. For the negative condition for X , if $\neg X \in H$ then $\neg(A \supset B) \in H$ and, since H is a Hintikka set, $A \in H$ and $\neg B \in H$. Since both A and B meet $(\star)$, $v(A) = \mathbf{t}$ and $v(B) = \mathbf{f}$. Then $v(X) = v(A \supset B) = v(A) \supset v(B) = \mathbf{t} \supset \mathbf{f} = \mathbf{f}$.

3.4.2 Completeness, Constructively

We will show that if a formula does not have a classical propositional tableau proof, then it has a counter-model, a boolean valuation that falsifies it. Thus either a formula has a counter-model, or it has a tableau proof, and indeed we will show that it has a proof in which all branches close because of contradictions *at the atomic level*. In fact, we present two different kinds of completeness proofs. One is constructive, in the sense that it gives a method by which we can either find an atomically closed proof for a given formula, or find a counter-model. Thus we have a decision procedure. Of course, truth tables also provide a decision procedure, but tableau methods carry over to many common modal logics, while truth tables don't. The other proof, given following, doesn't have this nice decidability feature.

It shows that if a formula doesn't have a proof it will have a counter-model, but doesn't tell us how to find it. On the other hand, this kind of completeness proof is easier to describe, and continues to be easier to describe as logics become more complex and constructive methods are harder to follow. This style of completeness is very similar to the axiomatic completeness argument given in Sect. 2.8, so we already know something about it.

Tableau rules are applied non-deterministically. We often find ourselves in the situation where more than one rule could be applied, and the choice of which is up to us. But we might have some algorithm we could follow, telling us in what order to carry out the rule applications. Such an algorithm is called *fair* if each rule application that could be made eventually is made. Of course we don't, in general, care about a rule application that might be made on a branch that is closed—it has no use for us. There is, however, one exception—we would like to show that every tautology has an *atomically* closed tableau, so we will allow continued work on a branch that is closed, but not atomically so.

Definition 3.4.3 (Fair Tableau Construction Algorithm) A *fair* algorithm for the construction of tableaus is one that proceeds in stages such that, at each stage, for each atomically unclosed branch at that stage, and for each tableau node on the branch labeled with a formula to which a tableau rule can be but has not yet been applied on that branch, at some later stage the algorithm applies the appropriate tableau rule to the formula at that node on that branch.

It is not hard to show that fair tableau construction algorithms exist. Here is a sketch of one. At each stage of the construction pick a tableau branch that is not atomically closed, say the first one from the left to be definite, pick a formula on it that is not atomic, not the negation of an atomic formula, and that has had no rule applied to it on that branch, say the uppermost such formula to be definite, and apply the appropriate rule to it on that branch. If there are no usable formulas on the branch, move on to the next branch on the right. If there are no more branches, stop. It can be shown that this process must terminate, but we omit the proof here.

We now have all we need to establish completeness for tableaus, and to do so constructively. Let X be a propositional formula. Start a tableau for $\neg X$. That is, we have a one branch tableau with $\neg X$ at the root node, which is the only node on the branch. Suppose we have a fair tableau construction algorithm, and we apply it starting with this tableau, until the process stops because there are no more unused formulas on any open branch. We now have two possibilities. First, the result is a closed tableau, in which case we have a proof of X . Second, the result is a tableau that is not closed, but for which no further rule applications will add anything new. It is easy to see that for any open branch of this tableau, the set of formulas on it is a Hintikka set. This set will be satisfiable by Proposition 3.4.2—all its members map to **t** under some boolean valuation. But the Hintikka set must contain $\neg X$, since this formula is at the tableau root, and thus is on every branch. Then there is some boolean valuation mapping $\neg X$ to **t**, and hence mapping X to **f**, and so X has a counter-model. We have thus proved the following.

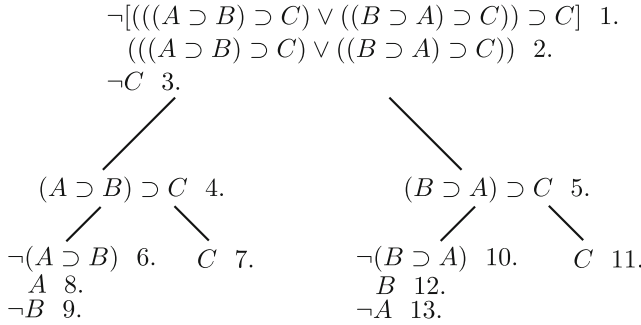
Theorem 3.4.4 (Tableau Completeness, Constructively) *For any propositional formula X , any fair tableau construction algorithm that begins with $\neg X$ will either produce an atomically closed tableau, thus proving X , or will yield information that can be used to produce a counter-model to X .*

The following example illustrates how this works. We use the fair construction algorithm sketched above: work from left to right, and from top down.

Example 3.4.5 Below is an attempted proof of

$$(((A \supset B) \supset C) \vee ((B \supset A) \supset C)) \supset C$$

using the algorithm sketched above. In it, 2 and 3 are from 1 by Rule $\neg \supset$, 4 and 5 are from 2 by Rule $\vee$, 6 and 7 are from 4 by Rule $\supset$, 8 and 9 are from 6 by Rule $\neg \supset$, 10 and 11 are from 5 by Rule $\supset$, and 12 and 13 are from 10 by Rule $\neg \supset$. The branches ending with 7 and 11 are atomically closed.



There are two unclosed branches, and both give us boolean valuations that are counter-models for $(((A \supset B) \supset C) \vee ((B \supset A) \supset C)) \supset C$. The branch ending with 9 gives us the valuation v_1 for which $v_1(A) = \mathbf{t}$ and all other propositional letters map to $\mathbf{f}$. The branch ending with 13 gives us the valuation v_2 for which $v_2(B) = \mathbf{t}$ and all other propositional letters map to $\mathbf{f}$.

A simple concluding remark to this work. We noted earlier that it would be shown there is a closed tableau proving a formula Z if and only if there is an atomically closed tableau proving Z . Exercise 3.1.3 asked you to show this directly. We now have a simple, indirect proof available. Of course trivially if Z has an atomically closed tableau proof, it has a closed tableau proof. Conversely, if Z has a closed tableau proof, by our soundness result, Theorem 3.3.4, Z is a tautology. And then by our completeness result, Theorem 3.4.4, Z will have an *atomically* closed tableau proof.

3.4.3 Tableau Completeness, Non-constructively

Since a constructive completeness proof for tableaux has just been given, one might wonder why we are about to discuss a non-constructive version that does not produce explicit counter-models. The reason is that as we move to more and more complex logics in the course of this book, tableau construction algorithms become more and more difficult to describe. But the non-constructive argument we are about to give continues to work, with straightforward changes as things get complicated. Presenting it in the relatively simple classical setting gives us a good background for things to come. It is very similar to the completeness proof given in Sect. 2.8 for a classical axiom system. In particular, note Definition 2.4.3.

Definition 3.4.6 (Tableau Consistency) Let S be a set of propositional formulas. If S is finite, we say S is *tableau consistent* if there is no atomically closed tableau beginning with the members of S . If S is infinite, we say it is *tableau consistent* if every finite subset is. S is *maximally tableau consistent* if it is tableau consistent, but no proper extension is tableau consistent.

A version of Proposition 2.4.2 is almost immediate. It says any extension of a tableau inconsistent set is tableau inconsistent, and any subset of a tableau consistent set is tableau consistent. We omit the simple proof.

Recall Lindenbaum's Lemma, Proposition 2.4.7. Almost the same proof gives us the following.

Theorem 3.4.7 (Lindenbaum's Lemma for Tableaus) *Let S be any tableau consistent set. Then $S \subseteq S_\infty$ for some maximally tableau consistent set S_∞ .*

Proof The proof of Proposition 2.4.7 showed how to extend an axiomatically consistent set S to a set S_∞ , where this was the limit of a chain of consistent sets, $S = S_1 \subseteq S_2 \subseteq S_3 \subseteq \dots \subseteq S_\infty$, where "limit" means "chain union". Exactly the same construction is used now, except that "consistent" is replaced with "tableau consistent". We do not repeat it all here. Further S_∞ , using tableaux, has the same basic features that the axiomatic version did, as follows.

First, S_∞ must be tableau consistent. For, if it were not, there would be a closed tableau beginning with some finite subset. All the members of this finite subset must be present in S_n for some number n , and so S_n would be not tableau consistent, but each S_n is.

Second, the tableau consistent version of S_∞ is maximally tableau consistent. And for this too the argument is as before, with "tableau consistent" replacing "consistent".

A set that is maximally consistent in the axiomatic sense has properties such as: it contains $X \vee Y$ if and only if it contains at least one of X or Y . For maximal *tableau* consistency, we cannot prove such a result at this point. What we can do easily is show that *if* such a set contains $X \vee Y$ *then* it contains one of X or Y . The converse implication is true, but we can't prove it yet without considerable effort.

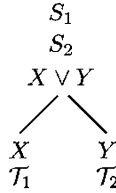
Fortunately, we don't really need the stronger result to establish completeness for tableaux. Here is what we can get with minimum effort.

Proposition 3.4.8 *Let S be a maximally tableau consistent set. Then S is a Hintikka set.*

Proof Part 1 of Definition 3.4.1 is simple. S does not contain both A and $\neg A$ for atomic A because if it did, a tableau starting with the finite subset $\{A, \neg A\}$ of S would be immediately closed, and S would not be tableau consistent.

All the conditions in part 2 of Definition 3.4.1 should be proved, but we only do one of them. It is easy, and serves as a guide for proving the rest. Let S be a maximally tableau consistent set. We show that if S contains $X \vee Y$ then S contains one of X or Y . It is actually the converse that we work with. Suppose S is maximally tableau consistent, but does not contain either X or Y . We show it does not contain $X \vee Y$.

Since S does not contain X but is maximal, it must be that $S \cup \{X\}$ is tableau inconsistent. Then there must be a finite subset $S_1 \subseteq S$ such that there is a closed tableau, $\mathcal{T}_1$, beginning with the members of $S_1 \cup \{X\}$. Similarly there must be a finite subset S_2 of S such that there is a closed tableau, $\mathcal{T}_2$, beginning with the members of $S_2 \cup \{Y\}$. Of course $S_1 \cup S_2$ is also a finite subset of S . Now construct a tableau as follows.



We begin the tableau construction with the members of $S_1 \cup S_2$ and $X \vee Y$. Then we branch. On the left branch we have all of the members of $S_1 \cup \{X\}$, and we can continue this branch to closure by copying over the steps from the closed tableau $\mathcal{T}_1$ for $S_1 \cup \{X\}$. Similarly the right branch can be continued to closure by copying the steps from the closed tableau $\mathcal{T}_2$ for $S_2 \cup \{Y\}$. Thus the entire tableau closes, and this tells us that $S_1 \cup S_2 \cup \{X \vee Y\}$ is inconsistent, hence $S \cup \{X \vee Y\}$ is inconsistent. It follows that S , being maximally consistent, does not contain $X \vee Y$.

And now here is our second completeness proof, not as strong as Theorem 3.4.4 since the existence of a deciding algorithm is missing,

Theorem 3.4.9 (Tableau Completeness, Non-constructively) *If formula X is a tautology, then X has an atomically closed tableau proof.*

Proof Suppose X does not have an atomically closed tableau proof. Then there is no atomically closed tableau for $\neg X$ by definition, so the set $\{\neg X\}$ is tableau consistent. It extends to a maximally tableau consistent set, Theorem 3.4.7. This set is a Hintikka set, Proposition 3.4.8, which is satisfiable, Proposition 3.4.2, and since

it contains $\neg X$, there is a boolean valuation mapping $\neg X$ to **t**, and hence X to **f**. Then X is not a tautology.

3.4.4 Coda

Semantically, classical propositional logic is characterized using boolean valuations. Tautologies are the formulas that are always true. But in terms of formal proof systems, we now have two quite different mechanisms, axiomatics and tableaux. It is more than just good fortune that they both prove the same formulas—they were designed to do so. Suppose one wanted to prove the equivalence of tableaux and axiomatics, without going through semantics, using completeness and soundness theorems. A natural way of proceeding would be to show how to translate tableau proofs into axiomatic proofs and also the other way around. This is harder than it looks at first glance. In particular, in going from axiomatics to tableaux we have the rule of *Modus Ponens* to deal with. We would need to prove that if formula X has a tableau proof, and formula $X \supset Y$ also does, then Y has a tableau proof. It is possible to give an algorithm to convert tableau proofs of X and $X \supset Y$ into a tableau proof of Y but it is hard, and its discovery in Gentzen (1935), was a major starting point for what today is called *proof theory*. We do not discuss this important part of formal logic here.

If one is satisfied with simply showing that tableaux and axiomatics both prove the same formulas, without insisting on knowing explicit translations between the two systems, we already have that. By the soundness and completeness results of this chapter and of Chap. 2, both prove exactly the tautologies, so both prove the same formulas. We don't have algorithmic equivalence, but we do have important links with an intuitively natural semantics. This way of doing things sets the pattern for the rest of our book.

Exercises

Exercise 3.4.1 For each of the following formulas, either give a tableau proof or give a boolean valuation that falsifies it, constructed from a tableau proof attempt. Assume P , Q , and so on are propositional letters.

1. $(\neg P \supset Q) \supset ((P \supset Q) \supset Q)$.
2. $((P \supset Q) \wedge (Q \supset R)) \supset \neg(\neg R \wedge P)$.
3. $(P \supset (Q \supset R)) \supset ((\neg R \supset \neg Q) \supset \neg P)$.
4. $((P \supset Q) \supset P) \supset P$.
5. $(P \supset (Q \vee R)) \supset (P \supset (Q \wedge R))$
6. $((((A \supset B) \supset (\neg C \supset \neg D)) \supset C) \supset E) \supset ((E \supset A) \supset (D \supset A))$.

Exercise 3.4.2 Since every Hintikka set is satisfiable, Proposition 3.4.2, no Hintikka set can be contradictory. Prove this directly by showing, using Complete Induction, Proposition 1.3.1, that if H is a Hintikka set, it cannot contain both X and $\neg X$ for any formula X .

Exercise 3.4.3 Prove or disprove each of the following:

1. The intersection of two Hintikka sets is a Hintikka set.
2. The union of two Hintikka sets is a Hintikka set.

Exercise 3.4.4 A *literal* is an atomic formula or the negation of one. Let $\mathcal{H}$ be a collection of Hintikka sets all of which have the same members that are literals. Prove that the union of the members of $\mathcal{H}$ is a Hintikka set.

3.5 Strong Completeness and Compactness

The material in this section, and in a corresponding section in Part II, is something that is not a central topic of this book, though it will make a small appearance from time to time. Nonetheless, it is important material and it would do you no harm to read the following through, even if you don't do the exercises.

We began the search for an axiomatization of propositional classical logic by introducing the notion of axiomatic consequence in Definition 2.2.1. Later, consequence was also characterized semantically in Definition 2.1.3, and in tableau terms in Definition 3.2.1. Our axiomatic characterization was the most general, since we began discussing it before we had selected any axioms or rules of inference, while the semantical and tableau versions assumed the full machinery for classical propositional logic was present. From here on in this section, when we talk about axiomatic consequence we will assume we have the full set of classical propositional machinery available, as given in Sect. 2.7. We thus have three distinct characterizations of something called *consequence* for classical propositional logic. Fortunately, we have the following, which can be seen as generalizing the soundness and completeness theorems so far.

Theorem 3.5.1 (Strong Soundness and Completeness) *Let S be a set of formulas and X be a single formula. For classical propositional logic, X is a semantic consequence of S , that is $S \models X$, if and only if X is an axiomatic consequence of S , if and only if X is a tableau consequence of S .*

If S is the empty set, the Theorem above is simply the soundness and completeness results for our axiom system, together with the soundness and completeness results for our tableau system. What is new is that S need not be empty, and we are talking about consequence and not simply validity (or being a tautology). Verifying this amounts to looking at our earlier proofs and suitably generalizing them. This is an excellent series of exercises that provides a good test of your understanding of the central proof theoretic ideas introduced earlier. We would not dream of taking

away the pleasure of discovering how to do this for yourselves, and so we leave the work to you.

When the work just outlined is done, what we will have is a *non-constructive* proof that tableaux and axiomatics accomplish the same thing. It is non-constructive because we show this by showing that both simply prove the tautologies and both agree on consequence. We will not have any method of turning a proof in one system into a proof in the other. Producing such a translation is harder than it looks at first glance, as we discussed briefly in the Coda at the end of Sect. 3.4. We say no more about this here.

Earlier we proved a monotonicity result: if S has X as a consequence (in any of our senses) and $S \subseteq S'$, then S' has X as a consequence. The proof of this is easy. Semantically it is Exercise 2.1.5. Axiomatically it is the proof of Proposition 2.2.2. Using tableaux it is Exercise 3.2.2. Indeed, Theorem 3.5.1 tells us that if we have any one of these, we automatically have all three.

There is also a very important *compactness* result: if S has X as a consequence, then X is also a consequence of some finite subset of S . It is possible to prove this for the semantic version of consequence, the axiomatic, and the tableau. But proving it directly for the semantic version is genuinely hard. Fortunately, it is extremely simple and intuitive for both the axiomatic and the tableau versions. Here is the axiomatic version; the tableau version is equally easy. If X is a consequence of S using our axiom system, there is a derivation of X from S . An axiomatic derivation has a finite number of lines, and so uses only a finite subset of S , which we can take to be the members of S' . Now combined with Theorem 3.5.1 we have the following semantic result—essentially a result about boolean valuations or truth tables.

Proposition 3.5.2 (Semantic Compactness) *For classical propositional logic, if $S \models X$ then there is some finite subset S_0 of S such that $S_0 \models X$.*

The following is also commonly referred to as a compactness result.

Corollary 3.5.3 *Let S be an infinite set of formulas, and suppose that for every finite subset S_0 of S there is some boolean valuation mapping every member of S_0 to $\mathbf{t}$. Then there is a boolean valuation that maps the entire of S to $\mathbf{t}$.*

Proof We show the contrapositive: if no boolean valuation can map the entire of S to $\mathbf{t}$, then this is also the case for some finite subset S_0 of S .

Suppose no boolean valuation maps the entire of S to $\mathbf{t}$. Then, quite trivially, we have $S \models (P \wedge \neg P)$. Then there is some finite subset S_0 of S such that $S_0 \models (P \wedge \neg P)$. Since no valuation can map $(P \wedge \neg P)$ to $\mathbf{t}$, no valuation can map all of S_0 to $\mathbf{t}$.

Not all logics that have some kind of semantic definition have a compactness property. But this is taking us too far from our main subject, and we leave things at this point.

Exercises

To establish part of Theorem 3.5.1 two soundness results are needed: if $S \vdash X$ either axiomatically or using tableaus, then $S \models X$. The axiomatic version was already given as Exercise 2.3.2, and the tableau version as Exercise 3.3.1. If you have not done these, consider doing so.

Exercise 3.5.1 Give a proof that if $S \models X$ then X is an axiomatic consequence of S using the axiom system of Sect. 2.7. Suggestion: use our axiomatic completeness argument from Sect. 2.8 as a guide. It is the special case where $S = \emptyset$. Begin by showing that if $S \not\models X$ axiomatically then $S \cup \{\neg X\}$ is consistent.

Exercise 3.5.2 Give a proof that if $S \models X$ then X is a tableau consequence of S . Generalize our non-constructive tableau completeness argument and show that, for any set S of formulas and any formula X , if $S \not\models X$ using tableaus, then there is a boolean valuation mapping all of S to **t** but X to **f**, and hence it is not the case that $S \models X$.

References

- D’Agostino, M., Gabbay, D., Hähnle, R., & Posegga, J. (Eds.). (1999). *Handbook of tableau methods*. Dordrecht: Kluwer.
- Fitting, M. (1996). *First-order logic and automated theorem proving* (1st ed.) (1990). Berlin: Springer.
- Fitting, M. (1999). Introduction. In M. D’Agostino, D. Gabbay, R. Hähnle, & J. Posegga (Chap. Introduction, pp. 1–43). Cambridge: Kluwer.
- Gasquet, O., Herzig, A., Said, B., & Schwarzentruher, F. (2014). *Kripke’s worlds*. Studies in Universal Logic. Basel: Birkhäuser.
- Gentzen, G. (1935). Untersuchungen über das logische schliessen. *Mathematische Zeitschrift*, 39, 176–210, 405–431. (English translation as *Investigation into logical deduction* in Szabo, 1969, pp 68–131).
- Howson, C. (1997). *Logic with trees*. Digital edition, 2005. New York: Routledge.
- Jeffrey, R. (1967). *Formal logic: Its scope and limits*. New York: McGraw-Hill.
- Nerode, A., & Shore, R. (1997). *Logic for applications* (2nd ed.). Berlin: Springer.
- Smullyan, R. M. (1968). *First-order logic* (Revised Edition, Dover Press, New York, 1994). Berlin: Springer.
- Smullyan, R. M. (2009). *Logical labyrinths*. Natick: A. K. Peters.
- Smullyan, R. M. (2014). *A beginner’s guide to mathematical logic*. Mineola: Dover.
- Szabo, M. E. (Ed.). (1969). *The collected papers of Gerhard Gentzen*. Amsterdam: North-Holland.

Part II

Propositional Modal Logic

This Part of the book begins our presentation of modal logic. We discuss the semantics and some of the proof methods for *propositional* logics. You may already be acquainted with the subject, but still you probably should browse through this Part to become familiar with our notation, and especially with the family of tableau systems that we will be building on in everything that follows.

Propositional modal logic is a vast subject, and our presentation is relatively minimal. There are many, many books, papers, and conference proceedings about this. There are infinitely many propositional modal logics. Among these the best known are those that are *normal*, and there are infinitely many of these. We look at only a handful. Some of them are quite common in the literature and have significant applications. Others are less so, but are still illustrative of basic ideas. Just as in our treatment of propositional classical logic, we present both axiom systems and tableau systems. It should be noted that among modal logics, axiom systems are much more general than tableaux. It just happens that both exist and are well behaved for the logics we discuss here (or perhaps it is not quite a coincidence after all).

Chapter 4

Modal Logic, an Introduction



For analytic philosophy, formalization is a fundamental tool for clarifying language, leading to better understanding of thoughts expressed through language. Formalization involves abstraction and idealization. This is true in the sciences as well as in philosophy. Consider physics as a representative example. Newton's laws of motion formalize certain basic aspects of the physical universe. Mathematical abstractions are introduced that strip away irrelevant details of the real universe, but which lead to a better understanding of its "deep" structure. Later Einstein and others proposed better models than those of Newton, reflecting deeper understanding made possible by the experience gained through years of working with Newton's model.

"Model" is the key word here. A mathematical model of the universe is not the universe. It is an aid to the understanding—an approximately correct, though partial, description of the universe. As the Newton/Einstein progression shows, such models can improve with time, thought, and investigation.

What we present in this book is a mathematical model of the *modal* aspects of language. We apply formal techniques to clarify the interactions of modal issues with other important notions such as identity, names, and definite descriptions. We intersperse our formal mathematical presentation with discussions of the underlying philosophical problems that elucidate the basic issues.

In order to present a logic formally, several separate items are necessary. We need a rigorously defined *language* of course, but by itself this is pure, uninterpreted syntax. To give meaning to formulas of a language, we need a *semantics* that lets us single out those formulas we are primarily interested in, the *valid* ones. We also want a notion of *proof*, so that we can write down compact verifications that certain formulas are valid. And finally, we must connect the notion of proof with the semantics, so we can be sure it is exactly the valid formulas that have proofs.

We assume you are familiar with all this for classical propositional logic, and also we have included a refresher on this material in Part I. There, *truth tables* provide the semantics—valid formulas are those for which every line evaluates to **t** (such formulas are generally called *tautologies*). You may have seen axiom systems for

classical propositional logic, or tableau (tree) proof systems, or natural deduction systems. Since modal notions are more complex than propositional ones, one expects modal semantics and proof systems to be correspondingly more complex. Such expectations will be fulfilled shortly.

4.1 What Is a Modal?

A modal qualifies the truth of a judgment. *Necessarily* and *possibly* are the most important and best known modal qualifiers. They are called “alethic” modalities, from the Greek word for *truth*. In traditional terminology, *Necessarily P* is an “apodeictic judgment,” *Possibly P* a “problematic judgment,” and *P*, by itself, an “assertoric judgment.”

The most widely discussed modals apart from the alethic modalities are the *temporal* modalities, which deal with past, present, and future (“will be,” “was,” “has been,” “will have been,” etc.), the *deontic* modalities, which deal with obligation and permission (“may,” “can,” “must,” etc.) and the *epistemic* or *doxastic* modalities, which deal with knowledge and belief (“certainly,” “probably,” “perhaps,” “surely,” etc.). Prior (1955), perhaps a bit tongue-in-cheek, quotes the eighteenth century logician Isaac Watts to show the variety of modals that are available:

We might also describe several *moral* or *civil Modes* of connecting two Ideas together (*viz.*) *Lawfulness* and *Unlawfulness*, *Conveniency* and *Inconveniency*, etc. whence we may form such *modal Propositions* as these: ... *It is lawful for Christians to eat Flesh in Lent* There are several other *Modes* of speaking whereby a Predicate is connected with a Subject: Such as, *it is certain*, *it is doubtful*, *it is probable*, *it is improbable*, *it is agreed*, *it is granted*, *it is said by the ancients*, *it is written* etc. all which will form other Kinds of *modal Propositions*. (pp. 215–216)

The modals capture what traditional linguists have categorized as *tense*, *mood* and *aspect*, but which, in more modern treatments of grammar, are lumped together as *adverbials*, i.e., as modifiers of adjectives or verbs, and, derivatively, of nouns. In the sentence-frame,

John is _____ happy

the blank space can be filled by any of the modals: *necessarily*, *possibly*, *contingently*, *known by me to be*, *believed by you to be*, *permitted to be*, *now*, *then*. Alternatively, these qualifiers can be inserted into the sentence-frame

It is _____ true that John is happy

and regarded as modifying the truth of the claim. Hence the two equivalent characterizations: the modal qualifies the predicate, or the modal qualifies the truth of a claim.

Not all modals give rise to modal *logics*, however. Usually there must be two modal operators that conform to the logical principles embodied in the Modal Square of Opposition (Sect. 4.3, Fig. 4.1). But this is to presuppose that a logic of the modals is even possible, a view that has been much debated in the philosophical literature. Before we introduce the Square of Opposition, then, we will discuss some of these issues.

Exercises

Exercise 4.1.1 Discuss the following.

1. Is ‘probably’ a modal?
2. Is ‘truly’ a modal?
3. Is ‘needless to say’ a modal?
4. Is ‘not’ a modal?

4.2 Can There Be a Modal Logic?

Aristotle set the tone for the older tradition of logic—a tradition that stretched from antiquity, through the medieval period, as far as Leibniz—by including the study of modality in his book on the syllogism:

Since there is a difference according as something belongs, necessarily belongs, or may belong to something else (for many things belong indeed, but not necessarily, others neither necessarily nor indeed at all, but it is possible for them to belong), it is clear that there will be different syllogisms to prove each of these relations, and syllogisms with differently related terms, one syllogism concluding from what is necessary, another from what is, a third from what is possible. *Prior Analytics* i. 8 (29^b29–35) in Ross (1928)

The modal syllogism never attained the near universal acceptance of Aristotle’s more familiar logic of categorical statements, partly because of serious confusions in his treatment (documented by Kneale and Kneale, 1962). The same problems infected the discussions of many who followed him. Indeed, Kneale (1962) quotes a medieval logician who warns students away from the study of modality if they are to retain their sanity! Modality was closely linked with theological speculation and the metaphysics of the schoolmen, both of which were largely repudiated by the Enlightenment. The combination of interminable confusions and discredited metaphysics has given sustenance to those who have urged that there cannot be a modal logic.

The primary claim of the opposition is that modality has no place in the content of a judgment, so that the modal is inert with regard to the logical connections between that judgment and others. Kant (1781) is representative. Here is what he says about

the category of the modals presented in his *Table of Judgments* in Book I of the *Transcendental Analytic*:

The *modality* of judgments is a quite peculiar function. Its distinguishing characteristic is that it contributes nothing to the content of the judgment (for, besides quantity, quality, and relation, there is nothing that constitutes the content of a judgment), but concerns only the value of the copula in relation to thought in general. (p. 10)

And Kant's assessment was echoed by Frege (1879):

What distinguishes the apodeictic from the assertoric judgment is that it indicates the existence of general judgments from which the proposition may be inferred—an indication that is absent in the assertoric judgment. If I term a proposition 'necessary,' then I am giving a hint as to my grounds for judgment. *But this does not affect the conceptual content of the judgment; and therefore the apodeictic form of a judgment has not for our purposes any significance.* (p. 4)

Frege (1879) laid the groundwork for the modern treatment of logic by providing a notation and a set of axioms and rules of inference for propositional, first-order, and higher-order logic. His decision to omit modality was an influential factor in delaying the contemporary development of modal logic.

Frege's remarks above are found in a section where he explains which distinctions he will attempt to capture in his logical symbolism, and which he will not. His *Begriffsschrift* (literally, *Concept Script*) was designed to capture *conceptual content*, "that part of judgments which affects the *possible inferences*. . . . Whatever is needed for a valid inference is fully expressed; what is not needed is for the most part not indicated either . . ." (p. 3) Compare, for example, "The Greeks defeated the Persians at Plataea" and "The Persians were defeated by the Greeks at Plataea." The active and passive forms of a judgment, he argues, each have exactly the same inferential connections—any judgment that entails (or is entailed by) the first entails (or is entailed by) the second. The grammatical distinction may therefore be of psychological significance, but it is of no logical significance, so the active/passive distinction is not formally marked. By contrast, there is a clear logical distinction between "The Greeks defeated the Persians at Plataea *and* Thermopylae," on the one hand, and "The Greeks defeated the Persians at Plataea *or* Thermopylae," on the other; and both conjunction and disjunction are therefore representable in his *Begriffsschrift*.

It is true, as Frege says, that the content represented by P remains the same whether preceded by "necessarily," "possibly," or nothing at all. But it does not follow that the apodeictic, problematic and assertoric judgments all have the same conceptual content. Contrary to what Frege says, modal distinctions do affect possible inferences. Let us avail ourselves of modern notation: $\Box P$ for *It is necessary that P* and $\Diamond P$ for *It is possible that P*. Now,

$$P \supset \Diamond P$$

(i.e., *It’s actual, so it’s possible*) is usually considered to be valid—Hughes and Cresswell (1968) call it the “Axiom of Possibility”—but its converse

$$\Diamond P \supset P$$

(i.e., *It’s possible, so it’s actual*) is not. By Frege’s own criterion, P and $\Diamond P$ differ in conceptual content. Frege’s logic does not capture these formal inferences, but they are inferences nonetheless, and there is no compelling reason why logic should not be widened to accommodate them. Can there be a modal logic, then? There is no *a priori* reason why not. The most effective way of showing that there can be a modal logic is to actually construct one. We will construct several!

4.3 Aristotle’s Modal Square

A rough minimal requirement for a modal logic is that $\Box$ and $\Diamond$ satisfy the Modal Square of Opposition given in Fig. 4.1. (Additional requirements have been suggested, but there is no generally accepted standard. See Lukasiewicz (1953); Bull and Segerberg (1984); Koslow (1992) for alternatives.)

The Modal Square is due to Aristotle, who worked out the basic logical connections between necessity and possibility in his *De Interpretatione*. In Chap. 12, he argued that the negation of

It is possible that P

(4.1)

is not “It is possible that *not* P,” but rather “It is *not* possible that P.” Similarly, he argued that the negation of “It is necessary that P” is not “It is necessary that *not* P,” but rather “It is *not* necessary that P.”

In Chap. 13, he connected up the two modalities by identifying

It is *not* necessary that *not* P.

(4.2)

Fig. 4.1 The modal square of opposition

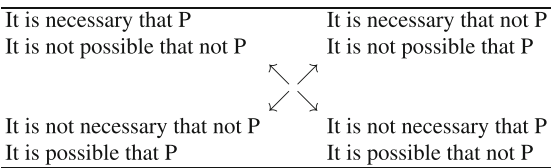
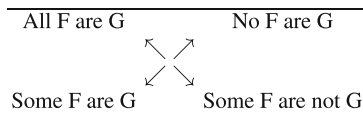


Fig. 4.2 The square of opposition for categorical statements



as the equivalent of (4.1). Aristotle's claim that (4.1) is equivalent to (4.2) reflects the standard (first half) of the interdefinability of the two modal operators:

$$\Diamond P \equiv \neg \Box \neg P$$

$$\Box P \equiv \neg \Diamond \neg P.$$

And we thereby get the Modal Square of Opposition given in Fig. 4.1.

The Modal Square of Opposition is analogous to the more familiar Square of Opposition for Categorical Statements given in Fig. 4.2. The schemas across the top row are *contraries* (cannot both be true), the schemas across the bottom row are *subcontraries* (cannot both be false), the schemas in each column are *subalternatives* (the top implies the bottom), and the schemas across the diagonals are *contradictories* (cannot have the same truth value).

In both Squares of Opposition, the schemas opposed on the diagonals are contradictory, but the other logical relations that hold among the categorical statements need have no analogue in the modal case. For example, whereas for the categorical statements, the top schema in a given column implies the bottom schema, the analogous modal schema need not imply the bottom one. For $\Box P$ to imply $\Diamond P$, the modal logic must be at least as strong as the modal logic called **D** (to be introduced later). Again, although the analogy between $\Box$ and $\forall$, and then also between $\Diamond$ and $\exists$ is secure, there are limits to how far they go. For example, although $\forall x Fx$ entails Fx (this is just Universal Instantiation), $\Box P$ entails P only in those modal logics in which what is called the **T** axiom holds.

4.4 Informal Interpretations

In Sect. 4.1, we characterized a modal as an adverbial that fills the blank in a sentence frame like

John is _____ happy.

The situation is actually somewhat more complicated. Adverbials differ in their favored placement. *Necessarily* must precede the adjective; *now* and *then* can occur either before or after the adjective; *here* and *there* must come after the adjective. These are peculiarities of English, and are independent of the underlying logical

sense. In what follows, we will find it helpful to standardize the frame, with the modifier coming after the adjective like this:

John is happy _____

The modal we insert qualifies John's happiness, opening up a spectrum along which modifications of his happiness can be compared. Let us flesh out the adverbial position as

John is happy under interpretation i .

Think of an interpretation i as a particular assignment of truth values to the nonlogical particles in the proposition, with the logical particles understood as the usual boolean functions. On the simplest reading of $\Box$ and $\Diamond$, a proposition is necessarily true if it comes out true for every interpretation—i.e., if it is a logical truth—and a proposition is possibly true if it comes out true for at least one interpretation—i.e., if it is consistent. Our modal propositions

$\Box$ John is happy

$\Diamond$ John is happy

therefore correspond respectively to the quantified propositions

$(\forall i)$ (John is happy under interpretation i)

$(\exists i)$ (John is happy under interpretation i)

“John is happy,” without a modal operator, is understood as “John is happy under interpretation i_0 ,” where i_0 is the interpretation corresponding to the real world.

It is more usual in modal logic to speak of a possible interpretation as a *possible world* or a *possible state*. The terminology and the idea that a necessary truth is true in all possible worlds is derived from Leibniz.

Let us see how this reading works for $\Box P \supset P$. If P is true in every possible world, then it is true in the actual world. So, this is, on our informal notion of necessity, a reasonable claim. $P \supset \Box P$, on the other hand, is not. Just because P is true in the actual world, it need not be true in *every* possible world.

Modern modal logic employs a modification of Leibniz's notion by taking into consideration the way in which the modal relates the possible worlds. This is called an *accessibility* relation. We will have much to say about accessibility, so for now one example must suffice: temporal worlds are ordered by their place in time. Take a world Δ to be accessible from a world Γ if Γ occurs no later in the temporal order than Δ . A proposition is *necessarily* true if it is true at every *accessible* possible world; a proposition is *possibly* true if it is true at some *accessible* possible world.

Let w_i be a possible world, and let w_0 be the actual world. Let $\mathcal{R}$ be the accessibility relation: $x\mathcal{R}y$ is read *y is accessible from x*. Then

$\Box$ John is happy

corresponds to

$$(\forall w_i)_{w_0\mathcal{R}w_i} (\text{John is happy at } w_i)$$

where we read the quantifier as, “for all w_i in the relation $\mathcal{R}$ to w_0 ,” or less formally, as, “for all w_i accessible from w_0 .”

Consider $\Box P \supset P$ again. Once accessibility relations are brought in, this says that if P is true at every accessible possible world, then it is true at the real world. For this to be valid, the real world must be accessible to itself. One might think that is a reasonable condition, but it isn’t always assumed. In Sect. 4.5, for instance, we will consider time based modal logics. Taking accessibility to mean *at some future time*, we would not have the real world accessible to itself. If we take it to mean *now or in the future*, we would have it. Both are reasonable to consider. In one case the formula $\Box P \supset P$ is not valid and in the other it is.

The notion of accessibility, introduced by Kripke (1963a,b), makes it possible to model a rich variety of modal logics. If we consider interpretations or worlds connected by an *accessibility relation*, then various algebraic constraints on the relation turn out to correspond to specific modal principles. The formal connection between these algebraic constraints and modal principles is discussed in Sect. 5.4.

Let us see informally how the accessibility relation enables us to extend our understanding of the modal operators. Consider, for example,

$$\Diamond P \supset \Box \Diamond P \tag{4.3}$$

Let $\Diamond P$ mean *It is conceivable that P*. Then (4.3) says that if there is a possible world we can conceive of from the actual one—i.e., an *accessible (conceivable) world*—in which P is true, then from every accessible (conceivable) world there is an accessible (conceivable) world in which P is true. Is this so? If P is conceivable, is it inconceivable that it is inconceivable that P ? It is reasonable to suppose that from the standpoint of a Greek disciple of Euclid, it is inconceivable that parallel lines meet; but this is certainly not inconceivable to us. So, from our perspective, it is conceivable that parallel lines meet, making the antecedent of (4.3) true; but there is at least one conceivable situation (putting ourselves in the sandals of an ancient Greek) from which it is inconceivable that parallel lines meet, making the consequent of (4.3) false. Presumably we can put ourselves into the Greek’s sandals but he cannot put himself into our shoes, because we know something about the past, but he knows nothing about the future. This means that, in the case of *conceivability*, (4.3) does not hold.

Once accessibility among worlds has been highlighted as the preeminent characteristic of modal logics, we can interpret the modal operators far afield from their

originally intended meanings. For instance, Pratt (1976), Harel (1984), and Harel et al. (2012) present *Dynamic Logic*. Suppose we are speaking about all possible execution states relevant to some non-deterministic computer program α , and we define an accessibility relation $s\mathcal{R}_\alpha t$ such that t is a possible next state of α when the current state is s . We obtain a multi-modal logic with modal operators for each computer program, and in which $\Box P$ is read as, “Every possible execution of α leads to a situation in which P is true,” and $\Diamond P$ read as, “It is possible to execute α reaching a situation in which P is true.” Hence one can generalize the subject called modal logic: it is an attempt to characterize talk of deformations or transitions or alternatives.

Exercises

Exercise 4.4.1 There are four truth-table definable functions of a single input. Use these in doing the following.

1. Show that we can define a truth-functional operator $\Box$ for which $\Box P \supset P$ always evaluates to **t** but for which $P \supset \Box P$ does not.
2. Show that if we also require that $\neg\Box P$ must also always evaluate to **t**, no truth-functional operator will do.

Exercise 4.4.2 Suppose every world is accessible to every other. Would (4.3) then be valid? Discuss informally.

4.5 Temporal Interpretations

The temporal interpretations of $\Box$ and $\Diamond$ are as old as the study of modality itself. In modern times it was the temporal semantics developed over a number of years by A. N. Prior that provided the clearest understanding of modal logic, until general possible world semantics was created. We have stronger intuitions about temporal worlds than we do about the more abstract and austere alethically possible worlds, and frequently we will find ourselves using the temporal interpretation as an aid in the understanding of modal concepts.

For Frege, a complete thought, a *Gedanke*, is an eternal, nonlinguistic object whose truth value never changes. Quine has a corresponding notion of an *eternal sentence*. The sentence

$$\text{John is happy} \tag{4.4}$$

is not an eternal sentence; it does not, *by itself*, express a complete thought. For, unless John is very unusual, there are periods in history in which it is true and periods in history in which it is false: it is not, by itself, timelessly true or false. If

we were to add a time determination to the sentence, we would stabilize its truth value. Let us, then, qualify *when* John is happy. The sentence frame is effectively filled with a time index t ,

$$\text{John is happy at time } t, \quad (4.5)$$

where t occupies a position that can be bound by a quantifier. We suppose there to be a range of times at some of which, perhaps, John is happy, and at others, not. Suppose (using an example from the first edition of this book) that John bought the latest model computer in November of 1994. Then, it is reasonable to suppose that the sentence

$$\text{John is happy in November 1994} \quad (4.6)$$

is true. Suppose, as always happened, that a more powerful computer is introduced some months later and the one he bought has dropped significantly in price. Then, it is reasonable to suppose that the sentence,

$$\text{John is happy in February 1995} \quad (4.7)$$

is false. We can existentially generalize on the true (4.6) to get the true

$$(\exists t) \text{ John is happy at time } t.$$

Given the falsity of (4.7), we have the falsity also of its universal generalization

$$(\forall t) \text{ John is happy at time } t.$$

Treating (4.4) as (4.5), in effect, is to regard *that John is happy* not as a proposition, but as a property of times. On this reading, (4.5) is an open sentence. When t is inside the scope of a quantifier, it is bound by the quantifier; when outside the scope of a quantifier, it acts as an indexical *now* denoting the designated (actual) time.

This gives us our first example of a modal logic with a temporal interpretation.

Example 4.5.1 This simplest temporal interpretation of $\Box P$ and $\Diamond P$, which parallels our alethic treatment of *necessarily* and *possibly*, takes

$$\Box \text{ John is happy}$$

$$\Diamond \text{ John is happy}$$

to correspond, respectively, to

$$(\forall t) \text{ John is happy at time } t$$

$$(\exists t) \text{ John is happy at time } t$$

As with the alethic interpretation, $P \supset \Diamond P$ turns out true and $\Diamond P \supset P$ turns out false; similarly, $\Box P \supset P$ turns out true and $P \supset \Box P$ turns out false.

Let us complicate our temporal interpretation by introducing the following future and past tense operators:

FP : It will sometime be the case that P

PP : It was sometime the case that P

where the verb in P is now untensed. It is common to define two more tense operators, **GP** ($\equiv \neg \mathbf{F}\neg P$) and **HP** ($\equiv \neg \mathbf{P}\neg P$):

GP : It will always be the case that P

HP : It has always been the case that P

We can, using these tense operators, capture a considerable number of natural language tenses. Let P be “John wins the election.” Then we have the following ways of interpreting natural language tenses:

- (a) P John wins the election
- (b) **FP** John will win the election
- (c) **PP** John won the election
- (d) **PPP** John had won the election
- (e) **FPP** John will have won the election
- (f) **PFPP** John would win the election

We will explain the iterated operators in (d)–(f) shortly.

We can develop two different modal logics from these operators by taking $\Box P$ and $\Diamond P$, respectively, to be **GP** and **FP**, or equally well, to be **HP** and **PP**. (The reader can verify that the Modal Square of Opposition is satisfied either way.)

These modal operators can be nested, and the complications thus introduced are nontrivial. We have, to this point, made no provisions for iterations of temporal operators. This is clear from our quantificational representation. If $\Diamond P$ says “There will be a time t such that John wins the election at t ,” then putting another temporal operator in front will appear to have no effect since there is no free variable left to bind. Yet, from the intuitive reading we give to the modal operators above, there are obvious distinctions: (e) is distinct from (f), for example, and (c) is distinct from (d).

The reason is that when we take into account the accessibility relation, we find that we have a free variable w_0 : $\Box F$ is understood to be $(\forall w_i)_{w_0 \mathcal{R} w_i} (F \text{ at } w_i)$. There is an implicit reference to the world in which the claim is made. If we place another modal operator in front of $\Box F$ to get $\Box \Box F$, we have

$$(\forall w_i)_{w_0 \mathcal{R} w_i} (\forall w_j)_{w_i \mathcal{R} w_j} (F \text{ at } w_j).$$

We still have our free variable w_0 for the world in which we make the claim. But, intuitively, we move from w_0 to w_i , then again from w_i to w_j , and when we get to w_j , that is where we evaluate F .

Let the present time be t . Then corresponding to the modal claims

F John wins the election,

P John wins the election,

G John wins the election,

H John wins the election,

we have the following respectively, where $t' < t$ means t' is an earlier time than t , while $t' > t$ means t' is a later time than t :

$$(\exists t')(t' > t \wedge \text{John wins the election at } t'),$$

$$(\exists t')(t' < t \wedge \text{John wins the election at } t'),$$

$$(\forall t')(t' > t \wedge \text{John wins the election at } t'),$$

$$(\forall t')(t' < t \wedge \text{John wins the election at } t').$$

Then (e) and (f) become, respectively,

$$(\exists t'')(t'' > t \wedge (\exists t')(t' < t'' \wedge \text{John wins the election at } t'))$$

and

$$(\exists t'')(t'' < t \wedge (\exists t')(t' > t'' \wedge \text{John wins the election at } t')).$$

(We leave (d) as an exercise for the reader.) So, (e) says that there will be a time in the future before which John wins the election: he will have won the election; and (f) says that there is a time in the past after which John wins the election: he would win the election.

Thus we have our second, more elaborate, temporal interpretation of the modal operators.

Example 4.5.2 Taking $\Box$ and $\Diamond$ to be **G** and **F**, respectively, gives us a different logic from the one in Example 4.5.1 even though the Modal Square of Opposition is still satisfied. For example, the thesis $\Box P \supset P$, fails. Just because it *will* always be true that P , it does not follow that P is true *now*. On the other hand, $\Box P \supset \Diamond P$ holds: if it will always be true that P , then there will be at least one (future) occasion at which P is true (assuming that time does not end). Other differences emerge when we consider iterations of the modal operators. Consider: $P \supset \Box \Diamond P$. This fails because from the fact that P is true, it does not follow that it will always be the case that there will be some further time at which it is true. On the other hand,

$\Box P \supset \Box\Box P$, holds because the temporal order is clearly transitive: if P holds at every future time, then at every future time P will hold at every (yet) future time.

Example 4.5.3 We can get yet another modal logic by defining $\Box$ and $\Diamond$ as follows:

$$\Diamond P = P \vee \mathbf{F}P$$

$$\Box P = P \wedge \mathbf{G}P$$

Thus $\Box P$ means P is and will remain the case; $\Diamond P$ means P is now or will at sometime be the case. (Prior (1957) claims that *Possibly* and *Necessarily* were used in this sense by the Megarian logician Diodorus.) In this system, $\Diamond P \supset \Box\Diamond P$ will fail. Suppose it either is or will be the case that P . Must this now and forever in the future be so? No. We might go too far into the future, to the point when P is a completed process. For instance, let P be “Scientists discover a cure for the common cold.” We hope that $\Diamond P$ is the case—someday a cure will be discovered. But once discovered, it cannot be discovered again, so $\Box\Diamond P$ is not the case.

There is an ordering relation that is imposed by time in temporal logic: *later than*. The temporal ordering is transitive, but it is neither reflexive nor symmetric. So, if we take $\Box P$ to be $\mathbf{G}P$, a proposition is necessarily true if it holds at all future worlds: a world, or state, is accessible if it is later in the temporal ordering. If, on the other hand, we take $\Box P$ to be $P \wedge \mathbf{G}P$, then a proposition is necessarily true if it holds now and forever later. $\Box P \supset P$ fails for the former reading, but it holds for the latter one.

Exercises

Exercise 4.5.4 What is the corresponding quantifier translation of item (d) $\mathbf{P}P$? Use our treatment of items (e) and (f) as models.

4.6 Historical Highlights

Now, armed with some formal machinery, it is time for a (highly selective) glance back in history at some well-known problems involving modal reasoning.

4.6.1 Aristotle’s Development of the Square

Aristotle had to fight hard to work out The Modal Square of Opposition we presented earlier in Fig. 4.1; confusions constantly threatened to obliterate the

Fig. 4.3 The modal square of opposition in *De. Int.* 13

It is necessary that P	It is necessary that not P
It is not possible that not P	It is not possible that P
It is not necessary that P	It is not necessary that not P
It is possible that P	It is possible that not P

gains. Aristotle opens *De Interpretatione* 13 with a somewhat different Square of Opposition, which we reproduce as Fig. 4.3. In this figure, the apodeictic forms have been transposed on the bottom row: in the lower left corner *It is possible that P* is alternately expressed as *It is not necessary that P*, and in the lower right corner *It is possible that not P* is alternately expressed as *It is not necessary that not P*. Aristotle had not clearly separated two readings of “possible” in *De Interpretatione* 13 (although he did so later at *Prior Analytics* 25^a37 – 40). Sometimes he takes it to mean *not impossible*; sometimes he takes it to mean *not impossible and not necessary*, i.e., *contingent*.¹ As Ackrill (1963) points out, the top half of the table appears to be driven by the first notion of “possible” while the bottom half appears to be driven by the second notion, i.e., of “contingent”.

Aristotle’s intention in setting up the logical relations as a Square of Opposition is that the forms in the left column are to be *subalternatives*, with the top implying the bottom. But while *It is necessary that P* may imply *It is possible that P*, it cannot imply the supposedly equivalent formulation *It is not necessary that P*. Aristotle saw the problem and sorted things out, eventually winning his way through to the square we gave in Fig. 4.1. He broke apart the two forms in the lower left hand corner. But his argument reveals a typical modal confusion that spills over to infect the relatively good grasp he had of nonmodal distinctions. At *De Interpretatione* 22^b10, he says:

For the necessary to be is possible to be. (Otherwise the negation will follow, since it is necessary either to affirm or to deny it; and then, if it is not possible to be, it is impossible to be; so the necessary to be is impossible to be—which is absurd.) (Ackrill, 1963, p. 63),

Aristotle’s conclusion, which we might express so

$$\Box P \supset \Diamond P, \quad (4.8)$$

is readily acceptable in any of the modal logics discussed that are at least as powerful as **D**. But the reasoning Aristotle put forward to justify (4.8) is faulty. To be sure, as he says, “it is necessary either to affirm or to deny it,” that is

Either (4.8) or not-(4.8).

But not-(4.8) ought to be

$$\neg(\Box P \supset \Diamond P)$$

¹ The former is known among the commentators as “one-sided possible,” the latter as “two-sided possible.”

Aristotle, however, concludes:

$$\Box P \supset \neg \Diamond P.$$

Ackrill (1963) regards this as a misapplication of the Law of Excluded Middle; but it is perhaps better viewed as a *scope* error. Aristotle confused the negation of the conditional with the negation of the consequent.

Exercises

Exercise 4.6.1 Can a Square of Opposition be constructed when $\Diamond$ is interpreted to mean “It is contingent that,” where *contingent* means true but not necessary?

4.6.2 Aristotle’s Future Sea Battle

The ancient Greek philosophers were much concerned with the relation between *necessity* and *determinism*. Events do not appear to occur haphazardly, but regularly, with earlier events affecting the course of later events. If an event B has its roots in an earlier event A, then it would seem that, given A, B could not help but occur; and if the earlier event A could not be altered, then it would be necessary, so we have not merely the *relative necessity* of B given A, but the *absolute necessity* of B. Everything, then, would seem to be occurring of necessity.

Aristotle meant by *necessity* “the unalterability of whatever has already happened” (Ackrill, 1963, 113). Now, *unalterability* appears to identify a fundamental asymmetry between past and future: the past is unalterable in a way the future is not. In one of the most famous arguments from antiquity, found in *De Interpretation* 9, Aristotle suppressed issues of causation and faced the problem of determinism in terms of *truth*, *truth at a given time* and *necessity*. One cannot undo what has already been done. But if it is true *now* that there will be a sea battle tomorrow, the future also appears determined and unalterable, and the asymmetry collapses:

Again, if it is white now it was true to say earlier that it would be white; so that it was always true to say of anything that has happened that it would be so. But if it was always true to say that it was so, or would be so, it could not be so, or not be going to be so. But if something cannot not happen it is impossible for it not to happen; and if it is impossible for something not to happen it is necessary for it to happen. Everything that will be, therefore, happens necessarily. So nothing will come about as chance has it or by chance; for if by chance, not of necessity. (18^b9 ff) in (Ackrill, 1963, pp. 50–51).

This is the famous problem of future contingents.

Aristotle clearly rejects the *fatalistic* or *deterministic* conclusion. “Absurdities,” he calls them: if “everything is and happens of necessity ... there would be no

need to deliberate ...” (18^b26 ff) And he offers the following solution beginning at 19^a23:

What is, necessarily is, when it is; and what is not, necessarily is not, when it is not. But not everything that is, necessarily is; and not everything that is not, necessarily is not. For to say that everything that is, is of necessity, when it is, is not the same as saying unconditionally that it is of necessity. Similarly with what is not. And the same account holds for contradictories: everything necessarily is or is not, and will be or will not be; but one cannot divide and say that one or the other is necessary. I mean, for example: it is necessary for there to be or not to be a sea-battle tomorrow; but it is not necessary for a sea-battle to take place tomorrow, nor for one not to take place—though it is necessary for one to take place or not to take place. So, since statements are true according to how the actual things are, it is clear that wherever these are such as to allow of contraries as chance has it, the same necessarily holds for the contradictories also. This happens with things that are not always so or are not always not so. With these it is necessary for one or the other of the contradictories to be true or false—not however, this one or that one, but as chance has it; or for one to be true *rather* than the other, yet not *already* true or false. (Akrill, 1963, 52–53).

Aristotle denies that events in the future are already determined. His response to the argument is to deny that statements about the future are in some sense already true or false. But there is considerable disagreement among commentators about what he meant by this. The text has subtle complexities that provide an interesting and instructive modal story.

Aristotle’s claim, “Everything necessarily is or is not,” introduces a possible source of confusion. For simplicity, we will take this to be a variant of the Law of Excluded Middle: $P \vee \neg P$. Depending upon the scope we assign to “necessarily,” we get two different claims. On the one hand, it might mean that the claim that P is true or $\neg P$ is true is necessarily true:

$$\Box(P \vee \neg P). \quad (4.9)$$

On the other hand, it might mean that P is necessarily true or $\neg P$ is necessarily true:

$$\Box P \vee \Box \neg P \quad (4.10)$$

Clearly, (4.9) is correct and (4.10) is problematic. If we maintain (4.10), then, together with *Excluded Middle*, the modality collapses, i.e., $P \equiv \Box P$: *Whatever is true is necessarily true; whatever is false is necessarily false*.

To be more precise, the modality collapses in a logic called **T**. We mentioned this logic at the end of Sect. 4.3, and we will be treating it formally starting with Sect. 5.4. In **T** we have, as validities, all formulas of the form $\Box X \supset X$ or, equivalently, $X \supset \neg \Box \neg X$. Here is an argument that, with the **T** assumption, we do have modal collapse. (4.10) says $\Box P \vee \Box \neg P$. By classical logic, $X \vee Y$ is equivalent to $\neg Y \supset X$. So, (4.10) gives us $\neg \Box \neg P \supset \Box P$. But in **T**, $P \supset \neg \Box \neg P$, so we have $P \supset \Box P$. And since $\Box P \supset P$ is valid in **T**, we have $P \equiv \Box P$.

It is not clear which of (4.9) or (4.10) Aristotle holds. In the passage quoted above, Aristotle appears to be clearly endorsing (4.9) but it is not as clear that he is rejecting (4.10). For, as Ackrill (1963) notes, the claim that “one cannot divide” might be understood to rule out either of the following inferences:

$$\frac{\Box(P \vee \neg P)}{\text{So, } \Box P} \quad \frac{\Box(P \vee \neg P)}{\text{So, } \Box \neg P}$$

This is not yet (4.10). Nonetheless, passages like this sustain one interpretation of the chapter, on which the deterministic conclusion is presumed to follow because of scope confusion. This interpretation, which Ackrill (1963) calls “the preferred interpretation,” sees Aristotle as arguing that the clearly and unproblematically true (4.9) is confused with the dubious (4.10), leading the unsuspecting modal novice to believe that since he is committed to (4.9), he is thereby committed to the *fatalistic* position.

This interpretation, however, does not place any special premium on statements about the future, and this appears to be the point Aristotle stresses in his solution. On a second interpretation, as Ackrill (1963) puts it, “Aristotle holds that a statement with a truth-value automatically has a necessity-value (if true, necessary; if false, impossible), but he claims that a statement may lack a truth-value and acquire one later.” (p. 140)

There are two ways of understanding Aristotle’s solution on this second interpretation. On the one hand, it could be that Aristotle accepts the following argument.

- (a) $P \vee \neg P$
- (b) $P \supset \Box P$
- (c) $\neg P \supset \Box \neg P$
- SO $\Box P \vee \Box \neg P$

But what he objects to is substituting **FP** for P in step (b) (i.e., *It will be the case that P*—see the discussion in Sect. 4.4). Unfortunately this interpretation is clouded by Aristotle’s remarks in the quotation above in which he appears to sanction the substitution of **FP** for P in $\Box(P \vee \neg P)$. An alternative view is that the argument requires the intermediate steps of ascriptions of truth to statements. Let **True P** mean *It is true that P*. Then, on this view, the argument goes

- (a) $P \vee \neg P$
- (b) $P \supset \text{True } P$
- (c) $\neg P \supset \text{True } \neg P$
- (d) $\text{True } P \supset \Box P$
- (e) $\text{True } \neg P \supset \Box \neg P$
- SO $\Box P \vee \Box \neg P$

Kneale and Kneale (1962) favor this latter reading, and they see Aristotle as rejecting steps (b) and (c): they distinguish the *Law of Excluded Middle*, i.e., $P \vee \neg P$, from the *Principle of Bivalence*, i. e., **True** $P \vee$ **True** $\neg P$, and they assert that Aristotle rejected the latter but not the former. Statements about future contingents lack truth values. More precisely, for Kneale and Kneale (1962), Aristotle holds that although *It is true that* there either will be or won't be a sea battle tomorrow, one cannot infer that either *it is true that* there will be a sea battle tomorrow or *it is true that* there won't be a sea battle tomorrow.

A related view, known as the *medieval interpretation*, was held by Boethius and Ammonius, among others. (See Gaskin (1995) for the full range of Aristotle's argument.) They took Aristotle's *true* to be *determinately* or *definitely true*. It is now definitely or determinately true that there either will be a sea battle tomorrow or not. But it is not now definitely true or determinately true that there will be a sea battle tomorrow nor is it now definitely true or determinately true that there won't be a sea battle tomorrow. The future contingent is not now determinately or definitely true (or false), so that the future is still open.²

Exercises

Exercise 4.6.2 Show that $\Diamond P \supset \Box P$ is not valid in **K**.

Exercise 4.6.3 We showed that the modality collapses in **T** if $\Box P \vee \Box \neg P$. Show that the argument must break down for **K**.

4.6.3 The Master Argument of Diodorus Cronus

Two schools of logic were distinguished in antiquity: the Peripatetic School, which followed the teaching of Aristotle, and the Stoic School, which was founded by Zeno. The most famous logical practitioner among the Stoics was Chrysippus, who was deeply influenced by Diodorus Cronus and by Philo. See Kneale and Kneale (1962) and Mates (1961).

Diodorus is responsible for one of the most famous of the Stoic arguments, the so-called "Master Argument." It is a matter of speculation as to *who* is the Master referred to—Diodorus or Fate. (Kneale and Kneale, 1962, 119) records the description from Epictetus:

The Master Argument seems to have been formulated with some such starting points as these. There is an incompatibility between the three following propositions, "Everything that is past and true is necessary," "The impossible does not follow from the possible," and

² This notion of *definite truth* has a formal representation using the notion of *supervaluations*. For a good discussion of supervaluations, see Fraassen (1966).

“What neither is nor will be is possible.” Seeing this incompatibility, Diodorus used the convincingness of the first two propositions to establish the thesis that nothing is possible which neither is nor will be true.

Sorabji (1980) notes that the first two premises can both be found in Aristotle, but not the conclusion: Aristotle believed that something could happen even though it never does—this cloak could be cut up even though it never is. There is good reason to think that Diodorus’s argument is a sharpened version of the fatalistic argument Aristotle had considered in *De Interpretatione* 9. We will now try to reconstruct Diodorus’s argument using the tools of modal logic.

Let us symbolize Diodorus’s first premise, *Everything that is past and true is necessary*, as

$$\mathbf{PP} \supset \Box \mathbf{PP}, \quad (4.11)$$

where $\mathbf{PP}$ is the tense-logical representation introduced in Sect. 4.5 for *It was the case that P*. There are a number of ways in which to interpret his second premise, *The impossible does not follow from the possible*. We shall take it as the Rule of Inference:³

$$\frac{P \supset Q}{\Diamond P \supset \Diamond Q} \quad (4.12)$$

We shall also need one more assumption that is not explicit in the argument, namely, that time is discrete.⁴ Then we can show that the specifically Diodorian claim follows: *What neither is nor will be is impossible*, i.e.

$$(\neg P \wedge \neg \mathbf{FP}) \supset \neg \Diamond P. \quad (4.13)$$

($\mathbf{FP}$ is the tense-logical representation for *It will be the case that P*).

To be concrete, let P be “There is a sea battle taking place.” Now, suppose there is no sea battle taking place and never will be, i.e.,

$$\neg P \wedge \neg \mathbf{FP} \quad (4.14)$$

If no sea battle *is* taking place and none *will*, then—and right here is where the assumption of the discreteness of time comes in—there was a time (for example, the moment just before the present) after which no sea battle would take place, i.e.,

$$(\neg P \wedge \neg \mathbf{FP}) \supset \mathbf{P}\neg \mathbf{FP} \quad (4.15)$$

³ Note that (4.12) is quite different from $(P \supset Q) \supset (\Diamond P \supset \Diamond Q)$: to infer $\Diamond P \supset \Diamond Q$ by the rule, we require not merely that $P \supset Q$ be true, but that it be logically true.

⁴ Prior (1967), whose reconstruction is similar to ours, also makes this very same assumption. In fact, all that is needed is backward discreteness: each instant has an immediate predecessor, which is a weaker assumption.

It follows by *Modus Ponens* from (4.14) and (4.15) that there was a time in the past when it was true that no sea battle would later take place, i.e.,

$$\mathbf{P} \rightarrow \mathbf{F}P. \quad (4.16)$$

So, by *Modus Ponens* and the first of Diodorus's premises, (4.11),

$$\Box \mathbf{P} \rightarrow \mathbf{F}P. \quad (4.17)$$

which is to say that it is not possible for it not to have been the case that it would be true that no sea battle will ever take place,⁵ i.e.,

$$\neg \Diamond \neg \mathbf{P} \rightarrow \mathbf{F}P. \quad (4.18)$$

So, $\neg \mathbf{P} \rightarrow \mathbf{F}P$ is impossible. But if the sea battle were taking place, it would not have been the case that no sea battle would ever take place, i.e.,

$$P \supset \neg \mathbf{P} \rightarrow \mathbf{F}P, \quad (4.19)$$

This is a logical truth based on the meaning of the tense operators. So, by our Rule of Inference (4.12),

$$\Diamond P \supset \Diamond \neg \mathbf{P} \rightarrow \mathbf{F}P, \quad (4.20)$$

And since we have the negation of the consequent in (4.18), an application of *Modus Tollens* yields

$$\neg \Diamond P. \quad (4.21)$$

On the assumption of (4.14) we have derived (4.21) using the Diodorian premises (4.11) and (4.12). Thus we have the Diodorian conclusion (4.13).

Diodorus's definition *The possible is that which either is or will be [true]* was apparently widely known throughout the ancient world,⁶ see Mates (1961); Prior (1957). And widely debated. It is none other than the *Principle of Plenitude*: every possibility is realized (at some time).

⁵ This is a good example of the virtues of symbolization: it is highly unlikely that we would have been able to come up with this English rendering so readily without the symbolic formulation.

⁶ His definitions of the remaining modals are entirely consistent:

The impossible is that which, being false, will not be true
The necessary is that which, being true, will not be false

4.6.4 The Once and Future Conditional

Some modern logicians read *If p, then q* as the *material conditional*: i.e., it is false if the antecedent is true and consequent false; otherwise it is true. Some logicians have also called it the *Philonean conditional*, after the Stoic logician Philo, who argued for this reading. Just as *necessity* was a source of heated dispute among the ancients, so too was the conditional. Mates (1961) reports that Diodorus disputed Philo's reading of the conditional, preferring to regard a conditional as true only when it is *impossible* for the antecedent to be true and consequent false—of course, in his special temporal sense of impossible, viz., that at no time is the antecedent true and consequent false. So, as (Mates, 1961, p. 45) puts it, “A conditional holds in the Diodorean sense if and only if it holds *at all times* in the Philonean sense.” How appropriate that the modern reintroduction of modal logic should turn on this very same issue.

Lewis (1918) is largely responsible for bringing modal logic back into the mainstream of logic.⁷ As with Diodorus, Lewis found the material conditional wanting as a formalization of *If p, then q*, especially insofar as it entailed the so-called Paradoxes of the Material Conditional. These are, first, that a false statement (materially) implies anything, i.e., $\neg P \supset (P \supset Q)$, and, second, that a true statement is (materially) implied by anything, i.e., $Q \supset (P \supset Q)$. Lewis sought to tighten up the connection between antecedent and consequent, and he introduced a new connective, symbolized by $\rightarrow$ (known as “the fishhook”). $P \rightarrow Q$, expresses that *P strictly implies Q*, and is definable in terms of $\Box$ as follows:

$$P \rightarrow Q =_{\text{def}} \Box(P \supset Q).$$

We might very well term this the *Diodorean conditional*, for to say that *P strictly implies Q* is to say that the material (i.e., Philonean) conditional holds necessarily. It is equivalent to $\neg\Diamond(P \wedge \neg Q)$: it is impossible for *P* to be true and *Q* to be false.

Lewis's introduction of $\rightarrow$ was, however, flawed by use/mention confusion. In *P strictly implies Q*, we appear to be speaking about propositions, not using them, and so *necessity* appears to be a predicate of sentences, not a propositional operator, as we have treated it in our formulation of modal logic. Lewis, unfortunately, was not

The nonnecessary is that which either is or will be false

Formally, these turn out as follows:

$$\begin{aligned}\Diamond P &\equiv P \vee \mathbf{F}P \\ \neg\Diamond P &\equiv \neg P \wedge \mathbf{G}\neg P \\ \Box P &\equiv P \wedge \mathbf{G}P \\ \neg\Box P &\equiv \neg P \vee \mathbf{F}\neg P\end{aligned}$$

⁷ Lewis eventually introduced five distinct formulations for modal logic, of which **S4** and **S5** are the most well known.

particularly careful in this regard. For example, he says in (Lewis and Langford, 1932, p. 247)

... the relation of strict implication expresses precisely that relation which holds when valid deduction is possible, and fails to hold when valid deduction is not possible. In that sense, the system of Strict Implication may be said to provide that canon and critique of deductive inference which is the desideratum of logical investigation.

Note the two-fold confusion in this passage. First, the expression “valid deduction” combines semantical and syntactical notions without clearly distinguishing them. Second, whether syntactical or semantical, the relation is one that holds between sentences: that the relation holds is a fact about sentences, and so the variables on either side of $\rightarrow$ are to be replaced by names of sentences (since one is speaking about them) rather than by sentences themselves. These confusions were seized upon by foes of formalized modal logic, and Lewis’s achievements were overshadowed in the subsequent controversy, a controversy which has its roots deep in the history of the subject, as we shall now see.

4.6.5 *The Reality of Necessity*

In order to understand the continuing controversy about modal logic, we must return to our discussion in Sect. 4.2 about whether the modal is part of the content of a judgment.

The cognitive content of a sentence, the proposition it expresses, is commonly thought of as encapsulated by its *truth conditions*, i.e., the features of the reality it purports to describe that make it true. The proposition that *John is a man*, for example, is true if and only if the individual *John* has the property of *being a man*. But the proposition that *John is necessarily a man* will require more for its truth than just that the individual *John* has the property of *being a man*—there will also have to be something in reality corresponding to his *necessarily* being a man. The something more is usually thought of as a *real essence*—a property an individual has necessarily, i.e., a property that is part of his essential nature so that he could not exist if he lacked the property. *Real essences*, *natural kinds*, *natural laws*, and related notions were characteristic of the scholastic thinking that derived from Aristotelian metaphysics. This prevalent way of thinking admitted necessary connections in reality, between an individual and some of its properties (real essences), between a group of individuals and some of their common properties (natural kinds), between an individual or group doing something and another (possibly the same) individual or group’s doing something (natural laws). From the Realist standpoint, a modal is comfortably conceived of in much the same way as any other part of a proposition, namely, as representing a feature of reality that determines the truth of the proposition.

One can readily understand, then, how a healthy skepticism about these necessities in nature will nudge one towards the Kant/Frege position described in Sect. 4.2 which denies that the modals belong to the content of a claim. The treatment of causality in Hume (1888) captures this view. A proposition saying that events of type A cause events of type B is widely thought to make a claim about the way things must be. Whenever events of type A occur, events of type B *must* occur. Hume, however, denies that there are necessary connections in nature. And if there are no necessary connections in nature, then no part of a proposition can represent them as anything that determines the truth of the proposition. So the “must” in the causal claim is not part of the proposition at all. Its role must be assigned elsewhere. Hume (1888) locates the seeming necessary connection between cause and effect in the habitual connection of our *ideas* of these causes and effects: the invocation of the idea of the cause immediately invokes the idea of the effect.

Despite their widely divergent views, Kant and Frege appear, like Hume, to have located necessity in the realm of ideas. (Kneale and Kneale, 1962, p. 36) say Kant holds “that a modal adverb such as “possibly” represents only a way of thinking the thought enunciated by the rest of the sentence in which it occurs.” In a similar vein, Frege says in his *Begriffsschrift*: “If a proposition is presented as possible, then either the speaker is refraining from judgment . . . or else he is saying that in general the negation of the proposition is false,” (Frege, 1952, p. 5). Both believe that the apodeictic and problematic indicators form no part of the content of the judgment. But the role they are supposed to play is a bit uncertain. One reading is that the modal introduces the speaker’s attitude toward the content, clearly locating it in non-cognitive psychology; another reading is that the modal connects the content with other judgments, moving it back into the realm of logic.

On the first reading, a judgment of possibility is a possible judgment. Appending “possibly” to a content indicates that one is merely entertaining it but not committing oneself to it. Or, it might have the force of “perhaps.” But this view seems correct only if one looks at assertions of modal claims. For, if a modal claim is not itself asserted, but forms part of a larger claim, occurring, for instance, as antecedent or consequent of a conditional like $(P \supset \Diamond P)$, this view no longer looks plausible.

It is quite possible that Frege (1879) came to believe that there is no difference between the assertoric and apodeictic judgments because he fell prey to a well-known modal error. Given the modal inference rule of *Necessitation* characteristic of *normal* modal logics,

$$\text{If } \vdash P \text{ then } \vdash \Box P \quad (4.22)$$

($\vdash$ symbolizes *it is asserted that*) and the “Axiom of Necessity,” as Hughes and Cresswell (1996) call it,

$$\Box P \supset P \quad (4.23)$$

we obtain the derived rule:⁸

$$\vdash P \text{ if, and only if } \vdash \Box P \quad (4.24)$$

And (4.24) makes it look as though there really is no logical difference in content between the assertoric and the apodeictic judgments. But, of course, this is not so. $\Box P$ and P differ in content. The true (4.24) must be distinguished from the (usually) false

$$P \equiv \Box P. \quad (4.25)$$

The collapse of the modality effected by (4.25) is avoided: although (4.23) (i.e., *If necessarily true, then true*) is true, its converse,

$$P \supset \Box P, \quad (4.26)$$

(i.e., *If true, then necessarily true*), is (usually) not.

On the second reading, the modal relates this content to other contents. As Frege said, to claim that a content is necessary is to say that “it follows only from universal judgments.” The necessity of a content appears to be a function of its justification. And this brings us back to the position of Lewis.

It should be quite clear to the reader how the philosophical issues concerning modality have polarized. The Realist about necessity has no problem taking a modal as part of the proposition. The Anti-Realist about necessity takes the modal out of the proposition and raises it up a level, either as a comment about the propositional representation or, noncognitively, as an expression of one’s attitude about the propositional representation. The Anti-Realist about necessity in the twentieth century has been most comfortable regarding modal claims, if cognitive at all, as explicitly about sentences, and so as *metalinguistic* claims, a level at which the development of any interesting modal logics is seriously restricted. In a very famous comment, (Marcus, 1992, p. 5) characterized the most serious contemporary foe of quantified modal logic, W. V. O. Quine, as believing “that it was conceived in sin: the sin of confusing use and mention.” But the real evil of quantified modal logic for Quine is its apparent commitment to “the metaphysical jungle of Aristotelian essentialism.” Quine, and his fellow critics of quantified modal logic, instead of arguing the merits, even the coherence of essentialism, drew battle lines around the coherence of modal logic itself. And this criticism was invariably that quantified modal logic rests on logical errors, like use/mention confusion, or illegal substitutions, errors that arise because meaningful elements of a sentence are taken to represent reality when no such representation can be taking place. The nature of the criticisms stems from underlying philosophical and metaphysical outlooks,

⁸ Equation (4.23), as we mentioned earlier, holds in **T**, but not in **K**. Even so, (4.24) holds in **K**. We leave this as an exercise.

from which point of view modal elements really don't belong in the content of a proposition (for they represent nothing real), and any logic that treats them as part of the content of a proposition is flirting with logical disaster. We will be revisiting these arguments throughout this book.

References

- Ackrill, J. K. (Ed.). (1963). *Aristotle's categories and De Interpretatione*. Oxford: Clarendon Press.
- Bull, R. A., & Segerberg, K. (1984). *Basic modal logic* (Chap. II–I, pp. 1–88). In D. M. Gabbay & F. Guentner, 1983–1989.
- Fraassen, B. C. V. (1966). Singular terms, truth-value gaps, and free logic. *The Journal of Philosophy*, 63, 481–495.
- Frege, G. (1879). *Begriffsschrift, eine der arithmetischen nachgebildete Formel-sprache des reinen Denkens*. Halle: Verlag von L. Nebert. Partial translation in Frege (1952).
- Frege, G. (1952). *Translations from the philosophical writings of Gottlob Frege*. P. Geach & M. Black (Eds). Oxford: Basil Blackwell.
- Gabbay, D. M., & Guentner, F. (Eds.). (1983–1989). *Handbook of philosophical logic*. Four volumes. Dordrecht: Kluwer.
- Gabbay, D. M., & Guentner, F. (Eds.). (2001 to present). *Handbook of philosophical logic* (2nd ed.). Springer. Multiple volumes. Dordrecht: Kluwer.
- Gaskin, R. (1995). *The sea battle and the master argument*. Quellen und Studien zur Philosophie; Bd 40. Berlin: Walter de Gruyter.
- Harel, D. (1984). *Dynamic logic* (Chap. 10, Vol. 2, pp. 497–604). In D. M. Gabbay & F. Guentner, 1983–1989. Dordrecht: D. Reidel.
- Harel, D., Kozen, D., & Tiuryn, J. (2012). *Dynamic logic* (vol. 4, pp. 99–217). In D. M. Gabbay & F. Guentner, 2001 to present.
- Hughes, G. E., & Cresswell, M. J. (1968). *An introduction to modal logic*. London: Methuen.
- Hughes, G. E., & Cresswell, M. J. (1996). *A new introduction to modal logic*. London: Routledge.
- Hume, D. (1888). *A treatise of human nature*. Oxford: Clarendon Press.
- Kant, I. (1781). *Critique of pure reason* (Trans.). Norman Kemp Smith, 1964. London: Macmillan.
- Kneale, W. (1962). Modality de dicto and de re. In *Logic, methodology and philosophy of science* (pp. 622–633). Stanford: North-Holland.
- Kneale, W., & Kneale, M. (1962). *The development of logic*. Oxford: Clarendon Press.
- Koslow, A. (1992). *A structuralist theory of logic*. Cambridge: Cambridge University Press.
- Kripke, S. (1963a). Semantical analysis of modal logic I, normal propositional calculi. *Zeitschrift für mathematische Logik und Grundlagen der Mathematik*, 9, 67–96.
- Kripke, S. (1963b). Semantical considerations on modal logic. *Acta Philosophica Fennica*, 16, 83–94.
- Lewis, C. I. (1918). *A survey of symbolic logic*. New York: Dover.
- Lewis, C. I., & Langford, C. H. (1932). *Symbolic logic* (2nd ed.). Mineola: Dover. 1959. The Century.
- Lukasiewicz, J. (1953). A system of modal logic. *The Journal of Computing systems*, 1, 111–149.
- Marcus, R. B. (1992). *Modalities*. New York: Oxford University Press.
- Mates, B. (1961). *Stoic logic*. Berkeley: University of California Press.
- Pratt, V. R. (1976). Semantical considerations on Floyd-Hoare logic. In *Proceedings of the 17th Symposium on the Foundations of Computer Science* (pp. 109–121). Piscataway: IEEE.
- Prior, A. N. (1955). *Formal logic*. Oxford: Clarendon Press.
- Prior, A. N. (1957). *Time and modality*. Oxford: Clarendon Press.

Prior, A. N. (1967). *Past, present and future*. Oxford: Clarendon Press.

Ross, W. D. (Ed.). (1928). *The works of Aristotle*. Oxford: Oxford University Press.

Sorabji, R. (1980). *Necessity, cause, and blame: perspectives on aristotle's theory*. Ithaca: Cornell University Press.

Chapter 5

Propositional Modal Logic



5.1 What Are the Formulas?

We now start the formal treatment of (propositional) modal logic. And we begin with syntax. The language of propositional modal logic extends that of classical propositional logic, Definition 1.2.1. Two new unary symbols are added, $\Box$ (necessarily) and $\Diamond$ (possibly). Just as $\wedge$ and $\vee$ are interdefinable using the De Morgan laws, $\Box$ and $\Diamond$ are interdefinable so one could just take one as basic. We generally take both, although in our axiomatics $\Box$ plays a more central role. Rather than just stating the new formula cases added to Definition 1.2.1 we give the definition in full for convenience.

Definition 5.1.1 (Propositional Modal Formula) The set of formulas is the smallest set meeting the following conditions:

1. Every propositional atom is a formula;
2. If X and Y are formulas, then $(X \wedge Y)$, $(X \vee Y)$ and $(X \supset Y)$ are formulas.
3. If X is a formula, then so is $\neg X$.
4. If X is a formula, so are $\Box X$ and $\Diamond X$.

Using this definition, for example, $((\Box P \wedge \Diamond Q) \supset \Diamond(P \wedge Q))$ is a formula. It can be read, “If P is necessary, and Q is possible, then it is possible that both P and Q .” Informally, we will still commonly omit the outer parentheses of a formula, or use a variety of parenthesis shapes, to make reading easier. Thus the formula we just gave may appear as $(\Box P \wedge \Diamond Q) \supset \Diamond(P \wedge Q)$ or even $[(\Box P \wedge \Diamond Q) \supset \Diamond(P \wedge Q)]$.

We are using $\Box$ and $\Diamond$ to symbolize modal operators. Some older books, following a different tradition, use L and M instead—Hughes and Cresswell (1996) is an example. In addition, when considering formal aspects of knowledge K is often used in place of $\Box$. If more than one person’s knowledge is concerned, K_a , K_b , etc. may be used—this is an example of a multi-modal logic. Likewise, modal operators corresponding to actions can be considered. For example, the action might

be that of going from New York to San Francisco. If $\Box$ is intended to denote this action, then if P is the proposition, “It is 1pm,” and Q is the proposition “It is after 2pm,” then $P \supset \Box Q$ would be read, “If it is 1pm, then after I go from New York to San Francisco (no matter how I do so), it will be after 2pm.” A modal logic intended for consideration of actions would also naturally be a multi-modal logic, since there are many different actions—generally the notation involves $\Box$ with action-denoting letters placed inside.

In this book we (generally) confine our presentation to just $\Box$ and $\Diamond$ since, if the formal aspects of these two are firmly understood, generalizations are easier to understand.

We stated a principle of Complete Induction, Proposition 1.3.1, which we used for proving syntactical facts about classical propositional formulas. Numbers come in via the notion of formula degree, Definition 1.3.2. That definition is now extended by adding modal operators to the unary case. Here is the modified version.

Definition 5.1.2 (Degree, Allowing Modal Operators) The degree of a propositional modal formula is the total number of occurrences of $\neg$, $\wedge$, $\vee$, $\supset$, $\Box$ and $\Diamond$ in it.

Exercises

Exercise 5.1.1 Using the definition verify that, in fact, $((\Box P \wedge \Diamond Q) \supset \Diamond \Box (P \wedge Q))$ is a formula.

Exercise 5.1.2 Prove that $((\Box P \wedge \Diamond Q) \supset \Diamond (P \wedge Q) \Box)$ is not a formula. Complete induction will come in here.

5.2 What Are the Models?

Although several formal semantics have been created for modal logics, *possible world* semantics has become the standard. It was introduced in Kripke (1963a,b, 1965) and today is commonly called *Kripke semantics*. Prior to Kripke, others had significant parts of the final modeling, but all were missing one or another key item. The history is interesting, and a good discussion can be found in Goldblatt (2003, 2006), in Section 4.

The notion of possible world is of a piece with standard model-theoretic interpretations of classical logic. As Kripke (1980, p. 19) says:

The main and original motivation for the ‘possible world analysis’—and the way it clarified modal logic—was that it enabled modal logic to be treated by the same set theoretic techniques of model theory that proved so successful when applied to extensional logic.

A valid formula (i.e., logical truth) of classical propositional logic is one that comes out true for every *possible* assignment of truth values to the statement letters. This notion of alternative interpretations should not pose any more of a philosophical problem in modal logic than it does in classical logic. Put a slightly different way, in so far as there are philosophical difficulties with the notion of possible world, these are difficulties that are already to be found in understanding multiple interpretations for statements of classical logic.

The difficulty is exacerbated by the fact that we say in modal logic, “There is a possible world such that . . .,” and so we quantify over possible worlds. According to a criterion of ontological commitment put forward in Quine (1948), this means that we are committed to the existence of such things. But some of these are intended to be mere possibilities, not actualities, and so we appear to be committed to the existence of things we would not otherwise want to be committed to. It looks as though we have all of a sudden increased our store of existents. Not only do we have the way things are, but also the way things might be. Once again, however, we must underscore that the very same difficulty crops up in the classical case, for we speak there of alternative interpretations, and we consider *all* possibilities—all truth-table lines—and these are possibilities that cannot all hold simultaneously. Since the notion of possible world is useful and coherent, it is quite clear that the problem is not with the notion of possible world but with some of the philosophical theories about them. But now for the formal details.

Definition 5.2.1 (Frame) A frame consists of a non-empty set, $\mathcal{G}$, whose members are generally called *possible worlds*, and a binary relation, $\mathcal{R}$, on $\mathcal{G}$, generally called the *accessibility relation*. Since a frame consists of two parts, we follow general mathematical practice and say a frame is a pair, $\langle \mathcal{G}, \mathcal{R} \rangle$, so we have a single object to talk about.

Understand that, while *possible world* is suggestive terminology, it commits us to nothing. In the mathematical treatment of frames, possible worlds are any objects whatsoever—numbers, sets, points, goldfish, etc.

We will generally use Γ , Δ , etc. to denote possible worlds. If Γ and Δ are in the accessibility relation $\mathcal{R}$, we will write $\Gamma \mathcal{R} \Delta$, and read this as Δ is *accessible from* Γ , or Δ is an *alternative world* to Γ . Informally the idea is, if Γ is a possible world and $\Gamma \mathcal{R} \Delta$ then Δ is a possible alternative to Γ —if Γ is how things are, then Δ is a way they could have been.

Definition 5.2.2 (Model) A frame is turned into a modal model by specifying which propositional letters are true at which worlds. We use the following notation for this. A *propositional modal model*, or *model* for short, is a triple $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$, where $\langle \mathcal{G}, \mathcal{R} \rangle$ is a frame and $\Vdash$ is a relation between possible worlds and propositional letters. We say the model $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ is *based on* the frame $\langle \mathcal{G}, \mathcal{R} \rangle$. In a model, if $\Gamma \Vdash P$ holds, we say P is *true at the world* Γ . If $\Gamma \not\Vdash P$ does not hold we symbolize this by $\Gamma \not\Vdash P$, and say P is *false at* Γ .

Notation varies from author to author. Sometimes one sees, instead of a relation $\Vdash$, a function v that is like a boolean valuation except that it takes two inputs, a

possible world and a propositional letter, $v(\Gamma, P)$ for instance, and produces a truth value, **t** or **f** as output. Sometimes one sees an indexed family of boolean valuations, where the indexing is by possible worlds, $v_\Gamma(P)$ for example. None of this should cause any serious problems.

The relation $\Vdash$ is often read as “forces,” a usage which comes from proofs of the independence results of set theory. In Smullyan and Fitting (1996) modal logic is, in fact, used quite directly for this purpose, illustrating how close the connection can be.

Keep classical propositional logic in mind as partial motivation. There, each line of a truth table (or each boolean valuation) is, in effect, a model. And a truth table line is specified completely by saying which propositional letters are assigned **t** and which are assigned **f**. Modal models are more complex in that a *family* of such truth assignments is involved, one for each possible world in $\mathcal{G}$, and there is some relationship between these assignments (represented by $\mathcal{R}$, though we have not seen its uses yet). Now in the classical case, once truth values have been specified at the propositional letter level, they can be calculated for more complicated formulas. The modal analog of this is, we calculate the truth value of non-atomic formulas *but at each possible world*. The notational mechanism we use is to extend the relation $\Vdash$ from Definition 5.2.2 beyond the propositional letter level.

Definition 5.2.3 (Truth in a Model) Let $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ be a model. The relation $\Vdash$ is extended to arbitrary formulas as follows. For each $\Gamma \in \mathcal{G}$:

1. $\Gamma \Vdash \neg X \iff \Gamma \not\Vdash X$.
2. $\Gamma \Vdash (X \wedge Y) \iff \Gamma \Vdash X \text{ and } \Gamma \Vdash Y$.
3. $\Gamma \Vdash (X \vee Y) \iff \Gamma \Vdash X \text{ or } \Gamma \Vdash Y$.
4. $\Gamma \Vdash (X \supset Y) \iff \Gamma \not\Vdash X \text{ or } \Gamma \Vdash Y$.
5. $\Gamma \Vdash \Box X \iff \text{for every } \Delta \in \mathcal{G}, \text{ if } \Gamma \mathcal{R} \Delta \text{ then } \Delta \Vdash X$.
6. $\Gamma \Vdash \Diamond X \iff \text{for some } \Delta \in \mathcal{G}, \Gamma \mathcal{R} \Delta \text{ and } \Delta \Vdash X$.

We will give concrete examples in the next section, but first a few remarks. The first four clauses of Definition 5.2.3 say that, at each world, the familiar propositional connectives behave in the usual truth table (or boolean valuation) way. The last two clauses are what is new here. The first, 5, says necessary truth is equivalent to truth in all possible worlds, but limited to those that are accessible from the one you are dealing with. The second, 6, says possible truth is equivalent to truth at some accessible possible world.

The notation we are using, $\Gamma \Vdash X$, does not mention the model, only a possible world and a formula. This is fine if it is understood which model we are working with, but if more than one model is involved, the notation can be ambiguous. In such cases we will write $\mathcal{M}, \Gamma \Vdash X$, where $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ is a model. We take this to mean: X is true at the world Γ of the collection $\mathcal{G}$ of possible worlds of the model $\mathcal{M}$. We will suppress explicit mention of $\mathcal{M}$ whenever possible, though, in the interests of a simpler notation.

Exercises

Exercise 5.2.1 Suppose we think of $\equiv$ as a defined connective in the usual way: $(X \equiv Y)$ abbreviates $(X \supset Y) \wedge (Y \supset X)$. Show that, at each possible world Γ of a modal model, $\Gamma \Vdash (X \equiv Y)$ is equivalent to $[\Gamma \Vdash X \text{ if and only if } \Gamma \Vdash Y]$.

Exercise 5.2.2 Show that, at each possible world Γ of a modal model, $\Gamma \Vdash (\Box X \equiv \neg \Diamond \neg X)$ and $\Gamma \Vdash (\Diamond X \equiv \neg \Box \neg X)$.

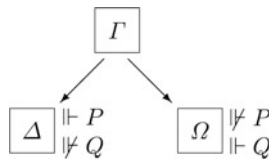
Exercise 5.2.3 Show that if a world Γ of a model has no worlds accessible to it, then at Γ every formula is necessary, that is, $\Box X$ is always the case, but no formula is possible, $\Diamond X$ is never the case.

5.3 Examples

In order to give a feeling for the behavior of modal models, we give several examples, and a few exercises. The pattern of presentation we follow is this. Models are given using diagrams, with boxes representing possible worlds. For two worlds of such a model, Γ and Δ , if $\Gamma \mathcal{R} \Delta$, we indicate this by drawing an arrow from Γ to Δ . We say explicitly that some propositional letters are true and some are false at particular worlds. If we do not mention a propositional letter, it is understood that its value doesn't matter for our example, it could be either true or false.

We begin with examples showing certain formulas *are not* true at worlds in a particular model, then move on to examples showing certain other formulas *are* true at worlds of models, under very broad assumptions about the models.

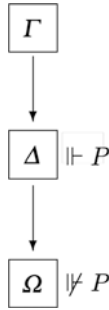
Example 5.3.1 Here is the first of our specific models.



In this model, $\Delta \Vdash P \vee Q$ since $\Delta \Vdash P$. Likewise, $\Omega \Vdash P \vee Q$. Since Δ and Ω are the only possible worlds in this model that are accessible from Γ , and $P \vee Q$ is true at both of them, $\Gamma \Vdash \Box(P \vee Q)$. On the other hand, we do not have $\Gamma \Vdash \Box P$, for if we did, since Ω is accessible from Γ , we would have $\Omega \Vdash P$, and we do not. Similarly we do not have $\Gamma \Vdash \Box Q$ either, and so we do not have $\Gamma \Vdash \Box P \vee \Box Q$. Consequently $\Box(P \vee Q) \supset (\Box P \vee \Box Q)$ is not true at Γ .

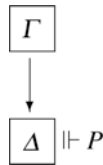
You might try showing $(\Diamond P \wedge \Diamond Q) \supset \Diamond(P \wedge Q)$ is also not true at Γ .

Example 5.3.2 This time we give a model in which $\Gamma \not\models \Box P \supset \Box\Box P$.

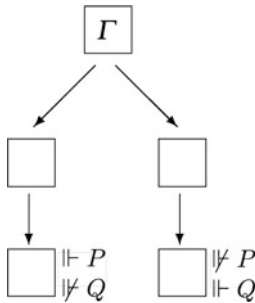


$\Gamma \models \Box P$ since $\Delta \models P$, and Δ is the only world accessible from Γ . If we had $\Gamma \models \Box\Box P$, it would follow that $\Delta \models \Box P$, from which it would follow that $\Omega \models P$, which is not the case. Consequently $\Gamma \not\models \Box P \supset \Box\Box P$.

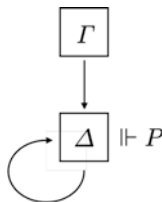
Example 5.3.3 Show for yourself that, in the following model, $\Gamma \not\models \Diamond P \supset \Box\Diamond P$.



Example 5.3.4 Again, show for yourself that in the following model, $\Gamma \not\models (\Diamond\Box P \wedge \Diamond\Box Q) \supset \Diamond\Box(P \wedge Q)$.



Example 5.3.5 This is a counter-example to lots of interesting formulas.



Since P is true at Δ , and Δ is the only world accessible from Δ , $\Box P$ is true at Δ . But then again, since $\Box P$ is true at Δ , it follows that $\Box\Box P$ is also true there, and likewise $\Box\Box\Box P$, and so on.

Now, since Δ is the only world accessible from Γ , and P is true at Δ , $\Box P$ is true at Γ . Likewise, since $\Box P$ is true at Δ , $\Box\Box P$ is true at Γ , as is $\Box\Box\Box P$, and so on. On the other hand, P itself is not true at Γ . Thus at Γ , all of the following formulas are false: $\Box P \supset P$, $\Box\Box P \supset P$, $\Box\Box\Box P \supset P$, and so on.

Next we turn to examples showing certain formulas must be true at worlds, under very general circumstances.

Example 5.3.6 In Example 5.3.1 we showed $\Box(P \vee Q) \supset (\Box P \vee \Box Q)$ could be false at a world under the right circumstances, so we cannot assume $\Box$ distributes over $\vee$. Now we show $\Box$ does distribute over $\wedge$, that is, $\Box(P \wedge Q) \supset (\Box P \wedge \Box Q)$ is true at every possible world of every model. This time we draw no pictures, since it is not a *particular* model we are dealing with, but the class of *all* models.

Suppose $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ is a model, and $\Gamma \in \mathcal{G}$. Also suppose $\Gamma \Vdash \Box(P \wedge Q)$; we show it follows that $\Gamma \Vdash \Box P \wedge \Box Q$.

Let Δ be an arbitrary member of $\mathcal{G}$ that is accessible from Γ , that is, $\Gamma \mathcal{R} \Delta$. Since $\Gamma \Vdash \Box(P \wedge Q)$, it follows that $\Delta \Vdash P \wedge Q$, and hence $\Delta \Vdash P$ and also $\Delta \Vdash Q$. Since Δ was arbitrary, at *every* possible world accessible from Γ both P and Q are true, hence at Γ both $\Box P$ and $\Box Q$ are true. But then, $\Gamma \Vdash \Box P \wedge \Box Q$.

In a similar way, it can be shown that the converse, $(\Box P \wedge \Box Q) \supset \Box(P \wedge Q)$ is true at every world, as is $\Box(P \supset Q) \supset (\Box P \supset \Box Q)$. We leave these to you.

Example 5.3.7 A relation $\mathcal{R}$ is called *transitive* if $\Gamma \mathcal{R} \Delta$ and $\Delta \mathcal{R} \Omega$ always imply $\Gamma \mathcal{R} \Omega$. Suppose $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ is a model in which $\mathcal{R}$ is transitive. Then $\Gamma \Vdash \Box P \supset \Box\Box P$, for each $\Gamma \in \mathcal{G}$. We show this as follows.

Suppose $\Gamma \Vdash \Box P$; we must show $\Gamma \Vdash \Box\Box P$. And to show $\Gamma \Vdash \Box\Box P$, let Δ be any member of $\mathcal{G}$ such that $\Gamma \mathcal{R} \Delta$ and show $\Delta \Vdash \Box P$. And to show this, let Ω be any member of $\mathcal{G}$ such that $\Delta \mathcal{R} \Omega$ and show $\Omega \Vdash P$. But, $\Gamma \mathcal{R} \Delta$ and $\Delta \mathcal{R} \Omega$, and $\mathcal{R}$ is transitive, so $\Gamma \mathcal{R} \Omega$. Also, we began by supposing that $\Gamma \Vdash \Box P$, so it follows that $\Omega \Vdash P$, which is what we needed.

Exercises

Exercise 5.3.1 Continue Example 5.3.1 and show that $(\Diamond P \wedge \Diamond Q) \supset \Diamond(P \wedge Q)$ is not true at Γ .

Exercise 5.3.2 Complete Example 5.3.3 by showing that $\Gamma \not\Vdash \Diamond P \supset \Box\Diamond P$.

Exercise 5.3.3 Complete Example 5.3.4 by showing that $\Gamma \not\Vdash (\Diamond\Box P \wedge \Diamond\Box Q) \supset \Diamond\Box(P \wedge Q)$.

Exercise 5.3.4 Show the following are true at every possible world of every model.

1. $(\Box P \wedge \Box Q) \supset \Box(P \wedge Q)$
2. $\Box(P \supset Q) \supset (\Box P \supset \Box Q)$

5.4 Modal Logics, Semantically Defined

Ever since Kripke semantics was introduced one has been able to specify modal logics using possible world machinery, and the number of logics of interest has grown hugely. Some of these logics are studied because of intended applications, in philosophy of course, but also in computer science, linguistics, and a surprising range of other areas. Sometimes families of modal logics are studied purely because of their mathematical properties. Here we are only concerned with philosophical applications and this limits things quite a lot, though it still means we have a multiplicity of modal logics to think about. Everyone (probably) agrees that whatever “necessity” might mean, necessary truths have necessary consequences. But opinions differ about whether necessary truths are so by necessity, or contingently. And probably few indeed have thought much about whether a proposition whose necessity is possibly necessary is necessarily true, or even possible. Once iterated modalities arise, intuition and experience tend to be inadequate. So in the philosophical literature a range of formal modal logics have arisen, partly in an attempt to capture differences in the proposed behavior of iterated modalities.

In the first half of the twentieth century a small range of particular modal logics were defined and investigated axiomatically. Some of these are still of fundamental importance and these are what we will primarily work with. They are quite enough to allow us to explore the first-order issues that constitute the fundamental topics of this book. Before presenting the list of logics we will make use of, some important facts should be mentioned. Modal logics are commonly specified semantically by making use of frames, or more specifically *sets* of frames. A particular set of frames may be quite difficult to specify, but the ones used here are quite simple and were among the first to be singled out when possible world semantics appeared. All the logics we consider have both axiom systems and tableau systems. This is far from the case generally, but this is not a book about the complexities of the vast family of modal logics. It is about applying formal methods to some of the topics of concern to those with philosophical interests, and the well-behaved logics we work with are quite enough for this purpose.

Historically, modal logics in the literature were first specified axiomatically. One of the early successes of possible world semantics was the discovery that modal logics could be specified by using sets of frames meeting various special conditions. We want to emphasize this: conditions are placed on *frames* and one uses the entire set of frames meeting such a condition. Although models are what we deal with directly, frames play a central role.

Definition 5.4.1 (Modal Validity) A formula X is *valid in a model* $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ if it is true at every world of $\mathcal{G}$, that is, $\Gamma \Vdash X$ for every $\Gamma \in \mathcal{G}$. X is *valid in a frame* if

it is valid in every model based on that frame. And finally, if $\mathcal{S}$ is a class of frames, X is $\mathcal{S}$ -valid if X is valid in every frame in $\mathcal{S}$.

In this book we concentrate on just a few quite standard propositional modal logics. These generally pre-date Kripke semantics, and as we have noted several times, they were originally characterized axiomatically. Modal axiom systems are discussed in Chap. 6. Axiomatics often makes it clearer what principles a logic is intended to capture. For the semantic treatment that follows, we will need some bits of mathematical terminology.

Definition 5.4.2 A relation $\mathcal{R}$ on a set $\mathcal{G}$ is:

- REFLEXIVE if $\Gamma \mathcal{R} \Gamma$, for every $\Gamma \in \mathcal{G}$;
- SYMMETRIC if $\Gamma \mathcal{R} \Delta$ implies $\Delta \mathcal{R} \Gamma$, for all $\Gamma, \Delta \in \mathcal{G}$;
- TRANSITIVE if $\Gamma \mathcal{R} \Delta$ and $\Delta \mathcal{R} \Omega$ together imply $\Gamma \mathcal{R} \Omega$, for all $\Gamma, \Delta, \Omega \in \mathcal{G}$;
- SERIAL if, for each $\Gamma \in \mathcal{G}$ there is some $\Delta \in \mathcal{G}$ such that $\Gamma \mathcal{R} \Delta$;
- EUCLIDEAN if, for each $\Gamma, \Delta, \Omega \in \mathcal{G}$, $\Gamma \mathcal{R} \Delta$ and $\Gamma \mathcal{R} \Omega$ implies $\Delta \mathcal{R} \Omega$.

The first three items in the definition above, reflexive, symmetric, transitive, taken together, define an *equivalence relation*. (Indeed reflexive and euclidean together also define an equivalence relation.) Equivalence relations will be studied in more detail in Sect. 11.1. In mathematics these are extremely common things. It was an important discovery that various familiar modal logics turned out to correspond to frames characterized by equivalence relations and by the weaker relations that resulted by dropping some of the conditions. The *reflexivity* condition says that every item is related to itself, while the *seriality* condition says that every item in $\mathcal{G}$ should be related to something, but not necessarily to itself. Clearly reflexivity implies seriality, so seriality is the weaker condition of the two. The significance of the *Euclidean* condition was not discussed in Kripke's original papers on modal semantics, but its importance was seen early on. It can be thought of as a generalization of the first of the Common Notions of *Euclid's Elements* (Heath (1908)): "Things which are equal to the same thing are also equal to one another."

Particular combinations of these conditions on frames determine the most common modal logics and we introduce them below, giving their standard names. Combinations other than these play a lesser role in the work here, and are relegated to the following section.

Definition 5.4.3 The following table contains the frame conditions for the basic modal logics that we will primarily use, and the names by which they are commonly known.

Logic	Frame Conditions
K	none
T	reflexive
B	reflexive, symmetric
S4	reflexive, transitive
S5	reflexive, symmetric, transitive or reflexive, euclidean
D	serial

It is perhaps easiest to understand what philosophical principles the conditions in the table above are trying to capture by looking at the axiom schemes that correspond to them. We do this in Chap. 6. For the time being, just postpone questions about why we are interested in these conditions until modal axiomatics has been introduced.

Here is a brief sketch of the history behind the table above. **S4** and **S5** are among the most common propositional modal logics one encounters, and appeared in Lewis and Langford (1932). (There were also **S1**, **S2**, and **S3**, but **S1** is never used today, and **S2** and **S3** have rather specialized interests.) Note that the frame conditions for **S5** amount to saying that it uses an *equivalence relation* as its accessibility relation. This gives it a special status, as equivalence relations are extremely well behaved. Kripke’s treatment of **S5** appeared separately in Kripke (1963a) well before his more complex work on other logics (though it was all thought through at about the same time). **T** comes from Feys (1937) (see also Feys, 1965). It was created independently under the name **M** in von Wright (1951), where it was formulated differently but equivalently, though the equivalence was not shown until later. **K** was named after Kripke, being the weakest modal logic that could be characterized using his semantics. **B** was introduced in Becker (1930) and given the name “Brouwersche system” (hence **B**) because of a perceived connection with one of the basic principles of Intuitionism, which was the creation of the mathematician Brouwer. The name **D** stands for *deontic*. A deontic logic has to do with obligations and permissions; von Wright (1951) pioneered the formulation of deontic principles as a logic. In such a logic one can think of $\Box X$ as something like “ X is obligatory,” or perhaps better, “Acting to make X the case is obligatory.” See Ballarín (2021) for more on the history of the logics from Definition 5.4.3.

Validity in a class of frames was specified in Definition 5.4.1. We say a formula X is **S4** valid if it is valid in all reflexive, transitive frames, and similarly for the other cases given in Definition 5.4.3. In particular, being **K** valid simply means being valid in every frame, and thus in every model. We saw, in Example 5.3.6, that $\Box(P \wedge Q) \supset (\Box P \wedge \Box Q)$ is true at every world of every model, so it is **K**-valid. Of course it is also **T**-valid, **S4**-valid, and so on. In that same Example, $(\Box P \wedge \Box Q) \supset \Box(P \wedge Q)$ and $\Box(P \supset Q) \supset (\Box P \supset \Box Q)$ were also said to be **K**-valid, and verification was left as an exercise. Likewise in Example 5.3.7 it was shown that $\Box P \supset \Box \Box P$ is valid in any model whose frame is transitive. It follows that this formula is **S4**-valid, and **S5**-valid as well, since both these include transitivity among their frame conditions.

Exercise 5.4.3 asks you to show that $\Box P \supset P$ is **T**-valid and that $P \supset \Box \Diamond P$ is **B**-valid. Also, $\Diamond P \supset \Box \Diamond P$ is **S5**-valid.

If a formula X is **S4**-valid it is true in all reflexive, transitive frames, and so in all symmetric, reflexive, and transitive frames, and hence it is also **S5**-valid. Thus **S4** is a *sublogic* of **S5**. In a similar way, **B** is a sublogic of **S5**. Trivially, **K** is a sublogic of all the others. Finally, every reflexive relation is serial so **D** is a sublogic of **T**.

Every frame in **T** is also a frame in **D**, but the converse is not true. Still it is conceivable that, *as logics*, **D** and **T** are the same. That is, it is conceivable that the same formulas are valid in the two classes of frames. (In fact, there are examples of two distinct classes of frames that determine the same logic. What is called *Gödel-Löb logic* is such an example, but we will not study it here.) However, this is not the case for **D** and **T**. It is easy to see that $\Box P \supset P$ is **T**-valid. But the model in Example 5.3.5 is based on a **D**-frame, and $\Box P \supset P$ is not valid in it. Thus as logics **D** and **T** are different.

Exercises

Exercise 5.4.1 Show $\Box P \supset \Diamond P$ is valid in all serial models.

Exercise 5.4.2 Show that, in **K**, a formula P is valid if and only if $\Box P$ is valid. (Note (4.24).) This is tricky if one uses validity directly, but becomes easier when tableaux are available. See Exercise 7.1.3.

Exercise 5.4.3 Assume $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ is a model, and show the following.

1. If $\mathcal{R}$ is reflexive, $\Box P \supset P$ is true at every member of $\mathcal{G}$.
2. If $\mathcal{R}$ is symmetric, $P \supset \Box \Diamond P$ is true at every member of $\mathcal{G}$.
3. If $\mathcal{R}$ is both symmetric and transitive, $\Diamond P \supset \Box \Diamond P$ is true at every member of $\mathcal{G}$.
4. If $\mathcal{R}$ is euclidean, $\Diamond P \supset \Box \Diamond P$ is true at every member of $\mathcal{G}$.

Exercise 5.4.4 Pick any two logics from the table in Definition 5.4.3 and show they do not have the same validities.

Exercise 5.4.5 Prove that a frame $\langle \mathcal{G}, \mathcal{R} \rangle$ is transitive *if and only if* every formula of the form $\Box P \supset \Box \Box P$ is valid in it.

Exercise 5.4.6 A *Euclidean* condition on relations was among those in Definition 5.4.2, but we have not used it yet. Let $\mathcal{R}$ be a reflexive relation on a set $\mathcal{G}$. Show that $\mathcal{R}$ is Euclidean if and only if it is symmetric and transitive. This leads to a different (and quite common) characterization of **S5**, and we have shown both in Definition 5.4.3. If **S5** is characterized this way, weakening reflexivity to seriality leads to a new logic. It is called **KD45** and has useful applications that we will not go into here.

Exercise 5.4.7 There is a modal logic that is generally called **S4.3** (due to Dummett and Lemmon) which is characterized semantically by the class of frames $\langle \mathcal{G}, \mathcal{R} \rangle$

for which $\mathcal{R}$ is reflexive, transitive, and meets the following *linearity* condition: whenever a possible world Ω has two possible worlds accessible from it, that is, whenever we have $\Omega \mathcal{R} \Gamma$ and $\Omega \mathcal{R} \Delta$, then either $\Gamma \mathcal{R} \Delta$ or $\Delta \mathcal{R} \Gamma$.

1. Show that $\Box(\Box P \supset \Box Q) \vee \Box(\Box Q \supset \Box P)$ is valid in all **S4.3** frames.
2. Show that $P \supset \Box \Diamond P$ can be falsified in some **S4.3** frame.
3. Show that $\Diamond \Box(P \supset Q) \supset (\Diamond \Box P \supset \Diamond \Box Q)$ is valid in all **S4.3** frames.

5.5 The Modal Cube

The modal logics specified in Definition 5.4.3 are among the older ones to be studied, and are still very common in the literature. They are hardly the only modal logics that have been considered—far from it—but they are all we really need to make the points we wish to make in this book. Still, it is not much work to ‘complete the set’, so to speak. All previous logics here are characterizable using reflexivity, symmetry, transitivity, and seriality. One more such condition was given in Definition 5.4.2, that of being *Euclidean*, which we have not made much use of. All together, we have given five possible frame conditions. Since we have 5 conditions, we have 2^5 , or 32, ways of combining them. We have given names for only 6 of them. The full family is natural, well behaved, and as we have seen, contains logics of long-standing interest. Collectively, the full family is given the name *Modal Cube*. The word ‘cube’ simply refers to the usual diagram, shown in Fig. 5.1, charting the relationships between the logics, which looks something like a cube. We discuss it a bit here, but it does not play a really central role in what follows.

Actually, the modal cube does not contain 32 different logics. Exercise 5.4.6 asks you to show that reflexivity, transitivity, and symmetry together determine the same set of frames as reflexivity and Euclidian. We thus have two different ways of characterizing **S5**, and in fact there are more. Every reflexive relation is also serial, for the obvious reason that if every possible world is related to itself, then it is also related to something. So any combination of frame conditions containing both reflexivity and seriality characterizes the same logic as one with seriality dropped. Once such things have been taken into account, the number of distinct frame classes definable using conditions from Definition 5.4.2 shrinks from 32 to 15.

The logics specified in Definition 5.4.3 had accepted, historical names. Once possible world semantics became available it was easy to create new logics, and naming them all became a problem. Systems of naming grew up. In the version we use here, all logics in the modal cube have systematic names that start with **K**, followed by specifications of which conditions from Definition 5.4.2 are imposed. Thus, for instance, **KD4** appears in Fig. 5.1b. Looking at the table in Fig. 5.1a, we see that this is the class of frames that are serial and transitive. We also use **KD4** as a name for the logic this class of frames determines. Variations on this naming pattern are easy to find in the literature, but the differences should cause no trouble.

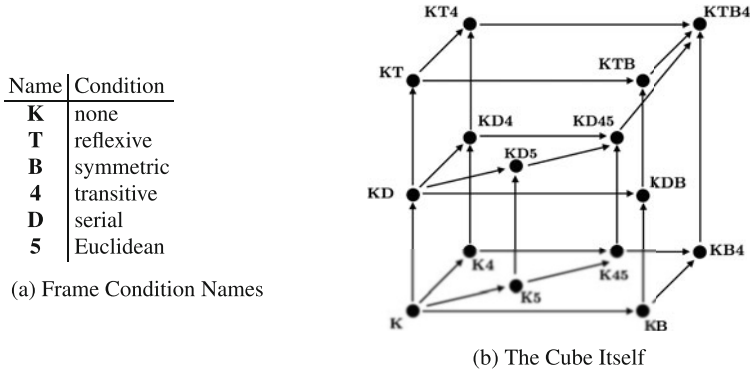


Fig. 5.1 The modal cube. (a) Frame condition names. (b) The cube itself

Standard Name	Systematic Name
K	K
T	KT
B	KTB
S4	KT4
S5	KTB4
D	KD

Fig. 5.2 Common names in the modal cube

In Fig. 5.1b we see a kind of three-dimensional coordinate system. The origin is the node labeled **K**. Think of the left-right axis as x , the depth axis as y , and the vertical axis as z . The arrangement is such that the strength of logics increases (there are more validities) as one moves in the positive x , y , or z direction. The x axis shows variability as **B** and **5** are added. The y axis covers **4** and **5**. The z axis shows variability among **D** and **T**. And finally, there are some lines that are diagonal. Two simultaneously modify x and y . They amount to alternate routes, and cover **4** and **5**. There is also one diagonal modifying x and z , and concerns **B**, **5**, **D** and **T**.

It was mentioned that sometimes more than one combination of conditions will determine the same class of frames. For instance, anything with **T** can have **D** added with no change. There are two more significant cases of duplications. The first involves **KB4**, which is the same as **KB45** and **KB5**. The second concerns **KTB4**, which is the same thing as any of **KT5**, **KTB5**, **KTB45**, **KT45**, **KTB4**, **KDB4**, **KDB45**, and **KDB5**, though we generally just call it by its common name, **S5**. Figure 5.2 shows where, in the cube, our common logics appear.

Exercises

Exercise 5.5.1 Show that **KT****B4** and **KT****45** determine the same class of frames.

Exercise 5.5.2 Show that **KB****5**, **KB****45**, and **KB****4** all determine the same class of frames.

5.6 Semantic Consequence

In logics generally, the notion of *follows from* is fundamental. Loosely speaking, one says a formula X follows semantically from a set of formulas S provided X must be true whenever the members of S are true. A precise version of this for classical logic is found in Definition 2.1.3, where the notation $S \models X$ is introduced for a set S of formulas and a single formula X . Here are two fundamental facts concerning classical semantic consequence.

MONOTONICITY If $S \models X$ and $S \subseteq S'$ then $S' \models X$;

COMPACTNESS If $S \models X$ then there is some finite subset S' of S such that $S' \models X$.

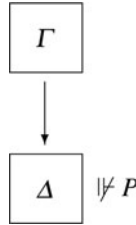
The first of these is easy to establish, and we left it for you as Exercise 2.1.5. A direct proof of the second is actually quite difficult, but one can approach it indirectly using soundness and completeness results (in particular, Theorem 3.5.1). Semantic consequence relates to axiomatic derivability, and an axiomatic derivation from a set S of formulas can only involve a finite number of formulas from S .

Things are more complex for modal logics. We want to keep the intuition that X is a logical consequence of S if X is true whenever the members of S are true, but a modal model can contain many possible worlds. Do we want to take X to be a consequence of S if X is true at each *world* at which the members of S are true, or do we want to take it as meaning X is valid in every *model* in which the members of S are valid? No such dichotomy occurs classically, but the choice presents itself modally, and the two notions are most decidedly *not* equivalent. As it happens, both versions are important, so we briefly sketch a treatment of them using notation from Fitting (1983). Further discussion can be found in Fitting (1993). The results below can be presented more generally, but we confine things here to the particular logics we have been discussing, for simplicity.

Definition 5.6.1 (Modal Consequence) Let $\mathbf{L}$ be one of the frame collections given in Definition 5.4.3. Also let S and U be sets of formulas, and let X be a single formula. We say X is a consequence in $\mathbf{L}$ of S as *global* and U as *local* assumptions, and we write $S \models_{\mathbf{L}} U \rightarrow X$, provided: for every $\mathbf{L}$ frame $\langle \mathcal{G}, \mathcal{R} \rangle$, for every model $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ based on this frame in which all members of S are valid, and for every world $\Gamma \in \mathcal{G}$ at which all members of U are true, we have $\Gamma \Vdash X$.

Thus $S \models_{\mathbf{L}} U \rightarrow X$ means X is true at those worlds of $\mathbf{L}$ models where the members of U are true, provided the members of S are true throughout the model, i.e., at every world. Admittedly, this is a more complex notion than classical consequence. Local and global assumptions play quite different roles, as the following examples make clear.

Example 5.6.2 We use $\mathbf{K}$ as our class of frames, the collection of all frames, we take S to be empty (so there are no global assumptions), U to contain only $\Box P \supset P$, and X to be $\Box\Box P \supset \Box P$. The following model shows $\emptyset \models_{\mathbf{K}} \{\Box P \supset P\} \rightarrow \Box\Box P \supset \Box P$ does not hold.



In this model, $\Gamma \not\models \Box P$ since $\Gamma \mathcal{R} \Delta$ and $\Delta \not\models P$. Consequently $\Gamma \models (\Box P \supset P)$, so the only member of $\{\Box P \supset P\}$ is true at Γ . On the other hand, $\Delta \models \Box P$ since there are no worlds alternate to Δ at which P fails. Since Δ is the only world accessible from Γ , it follows that $\Gamma \models \Box\Box P$. We have already seen that $\Gamma \not\models \Box P$, so $\Gamma \not\models (\Box\Box P \supset \Box P)$.

We thus have a counterexample to $\emptyset \models_{\mathbf{K}} \{\Box P \supset P\} \rightarrow \Box\Box P \supset \Box P$. Every global assumption is valid in the model since there are none. But at Γ all local assumptions are true and $\Box\Box P \supset \Box P$ is not.

Comparison of this example with the following will make clear the difference in behavior between local and global assumptions.

Example 5.6.3 This time we show that $\{\Box P \supset P\} \models_{\mathbf{K}} \emptyset \rightarrow \Box\Box P \supset \Box P$ does hold.

Let $\langle \mathcal{G}, \mathcal{R}, \models \rangle$ be a $\mathbf{K}$ model in which $\Box P \supset P$ is valid. Let Γ be an arbitrary member of $\mathcal{G}$ (at which all the local assumptions hold, since there are none). We show $\Gamma \models \Box\Box P \supset \Box P$.

Suppose $\Gamma \models \Box\Box P$. Let Δ be an arbitrary member of $\mathcal{G}$ such that $\Gamma \mathcal{R} \Delta$. Then $\Delta \models \Box P$. Since $\Box P \supset P$ is valid in the model, $\Delta \models (\Box P \supset P)$, and so also $\Delta \models P$. Since Δ was arbitrary, $\Gamma \models \Box P$.

The general properties of classical consequence carry over to modal consequence. For instance, classical consequence obeys a *monotonicity* condition, and so does the modal version.

Proposition 5.6.4 (Monotonicity) Suppose $S \models_{\mathbf{L}} U \rightarrow X$, and $S \subseteq S'$ and $U \subseteq U'$. Then $S' \models_{\mathbf{L}} U' \rightarrow X$.

Classical consequence has the *compactness* property, and this too carries over to a modal setting. As we noted above, the classical result can be proved directly but it is common to transfer the result from the context of formal proofs, where it is easy and obvious, to the semantical setting using completeness and soundness. We will do the same thing here.

Proposition 5.6.5 (Compactness) *Suppose $S \models_{\mathbf{L}} U \rightarrow X$. Then there are finite sets $S_0 \subseteq S$ and $U_0 \subseteq U$ such that $S_0 \models_{\mathbf{L}} U_0 \rightarrow X$.*

The classical logic Deduction Theorem also has a modal counterpart, but it is more complicated. Classically, the theorem says that $S \cup \{Y\} \models X$ if and only if $S \models (Y \supset X)$. Modally we have two versions, depending on whether Y is taken as a local or as a global assumption.

Proposition 5.6.6 (Local Deduction) *$S \models_{\mathbf{L}} U \cup \{Y\} \rightarrow X$ if and only if $S \models_{\mathbf{L}} U \rightarrow (Y \supset X)$.*

Proposition 5.6.7 (Global Deduction) *$S \cup \{Y\} \models_{\mathbf{L}} U \rightarrow X$ if and only if $S \models_{\mathbf{L}} U \cup \{Y, \Box Y, \Box\Box Y, \Box\Box\Box Y, \dots\} \rightarrow X$.*

The Local Deduction Proposition is essentially the same as the classical version. The Global version is obviously quite a different thing. Speaking informally, it says that having Y as a global assumption at a possible world Γ is equivalent to having Y at Γ , and having Y at every world that is accessible (that is, having $\Box Y$ at Γ), having Y at every world that is accessible from any world that is accessible (that is, having $\Box\Box Y$ at Γ) and so on. Then having Y as a global assumption at Γ is like having Y as a local assumption at every world we can reach from Γ via our accessibility relation, no matter how many times we use it. In a sense, it is the peculiar form of the Global Deduction Proposition that is at the heart of the complexities of modal consequence. In subsequent chapters we will need the notion of modal consequence from time to time, but it will almost always be with an empty set of local assumptions. Different books on modal logic sometimes take only local assumptions or only global assumptions as fundamental, and so the very definition of modal deduction can differ significantly from book to book.

Definition 5.6.8 We abbreviate $S \models_{\mathbf{L}} \emptyset \rightarrow X$ by $S \models_{\mathbf{L}} X$.

Exercises

Exercise 5.6.1 Show that $\emptyset \models_{\mathbf{K}} \{\Box P \supset P\} \rightarrow \Box\Box P \supset P$ does not hold, but $\{\Box P \supset P\} \models_{\mathbf{K}} \emptyset \rightarrow \Box\Box P \supset P$ does.

Exercise 5.6.2 Give a direct proof of Proposition 5.6.4.

Exercise 5.6.3 Let $\Box^n Y$ denote $\Box\Box\Box\Box\Box\Box\Box Y$, where we have written a string of n occurrences of $\Box$. Use the various facts given above concerning modal consequence

and show the following version of the deduction theorem: $S \cup \{Y\} \models_{\mathbf{L}} U \rightarrow X$ if and only if, for some n , $S \models_{\mathbf{L}} U \rightarrow (\Box^0 Y \wedge \Box^1 Y \wedge \dots \wedge \Box^n Y) \supset X$.

References

- Ballarín, R. (2021). Modern origins of modal logic. In E. N. Zalta (Ed.), *The Stanford encyclopedia of philosophy*. Stanford: Metaphysics Research Lab, Stanford University.
- Becker, O. (1930). Zur Logik der Modalitäten. *Jahrbuch für Philosophie und Phänomenologische Forschung*, 11, 497–548.
- Feys, R. (1937). Les Logiques Nouvelles des Modalités. *Revue Néoscholastique de Philosophie*, 40(56), 517–553.
- Feys, R. (1965). Modal logics. In J. Dopp (Ed.), *Collection de Logique Matheématique* (vol. 4). Louvain: E. Nauwelaerts.
- Fitting, M. (1983). *Proof methods for modal and intuitionistic logics*. Dordrecht: D. Reidel.
- Fitting, M. (1993). Basic modal logic. In D. M. Gabbay, C. J. Hogger, & J. A. Robinson (Eds.), *Handbook of logic in artificial intelligence and logic programming* (vol. 1, pp. 368–448). Oxford: Oxford University Press.
- Gabbay, D. M., & Woods, J. (Eds.). (2004 to present). *Handbook of the history of logic*. Amsterdam: North-Holland/Elsevier.
- Goldblatt, R. (2003). Mathematical modal logic: A view of its evolution. *Journal of Applied Logic*, 1, 309–392.
- Goldblatt, R. (2006). Mathematical modal logic: A view of its evolution. In D. Gabbay & J. Woods (Eds.), *Logic and the modalities in the twentieth century* (chap. 1, vol. 7, pp. 1–98). In Gabbay & Woods, 2004 to present.
- Heath, T. L. (1908). *The thirteen books of Euclid's elements* (2nd edn. 1925). Cambridge: Cambridge University Press. Reprinted by Dover Publications, 1956.
- Hughes, G. E., & Cresswell, M. J. (1968). *A new introduction to modal logic*. London: Routledge.
- Kripke, S. (1963a). Semantical analysis of modal logic I, normal propositional calculi. *Zeitschrift für mathematische Logik und Grundlagen der Mathematik*, 9, 67–96.
- Kripke, S. (1963b). Semantical considerations on modal logic. *Acta Philosophica Fennica*, 16, 83–94.
- Kripke, S. (1965). Semantical analysis of modal logic II, non-normal modal propositional calculi. In J. W. Addison, L. Henkin, & A. Tarski (Eds.), *The theory of models* (pp. 206–220). Amsterdam: North-Holland.
- Kripke, S. (1980). *Naming and necessity*. Cambridge: Harvard University Press. Reissued by Blackwell Publishing, 1991.
- Lewis, C. I., & Langford, C. H. (1932). *Symbolic logic* (2nd edn.) Mineola: Dover. 1959. The Century.
- Quine, W. V. O. (1948). On what there is. *Review of Metaphysics*, 2, 21–38. Reprinted in Quine, 1961.
- Quine, W. V. O. (1961). *From a logical point of view* (2nd. rev.). New York: Harper & Row.
- Smullyan, R. M., & Fitting, M. (1996). *Set theory and the continuum problem* (rev. edn.). Mineola: Dover. 2010. Errata at <http://melvinfitting.org/errata/errata.html>. Oxford University Press.
- von Wright, G. H. (1951). *An essay in modal logic* (L. E. J. Brouwer, E. W. Beth, & A. Heyting, Eds.). Studies in Logic and the Foundations of Mathematics. Amsterdam: North-Holland.

Chapter 6

Propositional Modal Axiom Systems



Historically, most of the best-known modal logics had axiomatic characterizations long before either tableau systems or semantical approaches were available. While early modal axiom systems were somewhat circuitous by today's standards, a natural and elegant system for **S4** was given in Gödel (1933), and this has become the paradigm for axiomatizing modal logics ever since. It is how we do things here.

Since an axiom system for classical propositional logic was presented in Chap. 2, we can build on it. In fact, we will just assume classical propositional logic is a known thing, and assert that some formula is a tautology and hence must have a proof, but we won't generally bother giving the proof. Our concern now is entirely on modal behavior.

6.1 The Logic $\mathcal{K}$ Axiomatically

We noted above that we will assume we know all about tautologies. In fact, we broaden the term somewhat. $(P \wedge Q) \supset P$ is a typical tautology. We can substitute complex formulas, possibly involving modal operators, for the propositional variables in such a tautology. For instance, we might substitute $\Box(X \vee Y)$ for P and $\Diamond X$ for Q in the tautology, getting $(\Box(X \vee Y) \wedge \Diamond X) \supset \Box(X \vee Y)$. *We will call the result of such a substitution a tautology too.*

Now here is the standard modal axiom system for the logic **K**, the simplest of the modal logics we consider.

Definition 6.1.1 (Axiom System for **K)** There are axiom schemes and rules of inference.

The axiom schemes for **K** are:

Classical Axioms All tautologies

Possibility $\Diamond X \equiv \neg\Box\neg X$, that is, $(\Diamond X \supset \neg\Box\neg X) \wedge (\neg\Box\neg X \supset \Diamond X)$

K Schema $\Box(X \supset Y) \supset (\Box X \supset \Box Y)$

The rules of inference for **K** are:

$$\text{Modus Ponens} \quad \frac{X \quad X \supset Y}{Y}$$

$$\text{Necessitation} \quad \frac{X}{\Box X}$$

It is important to note what the rule of necessitation does and does not say. It does not say $X \supset \Box X$ —if something is true, it is necessary. This is obviously undesirable. What it says is, if something is *provable*, it is necessary. That is quite different. Our axiom system captures a certain notion of logical truth, so if X is provable within the system, X is not simply true but must be a logical truth, and so we can conclude $\Box X$. It is the Necessitation Rule, as introduced by Gödel, that makes possible the simple, elegant modal axiom systems in use today. The following definition was seen before, in Sect. 2.2, but now it is in a modal context.

Definition 6.1.2 (Proof) An axiomatic *proof* is a finite sequence of formulas, each of which is either an axiom or else follows from earlier items by one of the rules of inference. An axiomatic *theorem* is the last line of a proof.

Before we give examples of proofs, it is convenient to give a certain *derived rule of inference*. By this is meant a rule of inference that can be safely added to our axiom system because applications of it can be “translated away.” That is, we have a standard way of replacing applications of the rule by uses of only the machinery that is an official part of the axiom system.

Definition 6.1.3 (Derived Rule of Regularity)

$$\frac{X \supset Y}{\Box X \supset \Box Y}$$

This is a derived rule because we can always replace an application of it in a proof with the following sequence of steps. From here on we will make free use of this rule.

$X \supset Y$	line occurring in a proof
$\Box(X \supset Y)$	Necessitation Rule on previous line
$\Box(X \supset Y) \supset (\Box X \supset \Box Y)$	Axiom K
$\Box X \supset \Box Y$	Modus Ponens on previous two lines

Now, here are two examples of proofs in the **K** axiom system. We have added line numbers and abbreviated explanations, though these are not officially parts of proofs.

Example 6.1.4 Axiom system proof of $\Box(X \wedge Y) \supset (\Box X \wedge \Box Y)$.

1	$(X \wedge Y) \supset X$	tautology
2	$\Box(X \wedge Y) \supset \Box X$	Regularity on 1
3	$(X \wedge Y) \supset Y$	tautology
4	$\Box(X \wedge Y) \supset \Box Y$	Regularity on 3
5	$[\Box(X \wedge Y) \supset \Box X]$ $\supset \{[\Box(X \wedge Y) \supset \Box Y]$ $\supset [\Box(X \wedge Y) \supset (\Box X \wedge \Box Y)]\}$	tautology
6	$[\Box(X \wedge Y) \supset \Box Y]$ $\supset [\Box(X \wedge Y) \supset (\Box X \wedge \Box Y)]$	Modus Ponens, 2, 5
7	$\Box(X \wedge Y) \supset (\Box X \wedge \Box Y)$	Modus Ponens, 4, 6

In the example above, line 7 comes from lines 2 and 4 using a common classical logic inference, embodied in the tautology $(A \supset B) \supset ((A \supset C) \supset (A \supset (B \wedge C)))$, an instance of which is line 5. *In the future we will simply say so in words, and abbreviate the formal proof.* We do this in the following; one of the classical tautologies involved but suppressed, for instance, is $(A \supset B) \supset ((B \supset C) \supset (A \supset C))$.

Example 6.1.5 Axiom system proof of $(\Box X \wedge \Box Y) \supset \Box(X \wedge Y)$.

1	$X \supset (Y \supset (X \wedge Y))$	tautology
2	$\Box X \supset \Box(Y \supset (X \wedge Y))$	Regularity on 1
3	$\Box(Y \supset (X \wedge Y)) \supset (\Box Y \supset \Box(X \wedge Y))$	axiom K
4	$\Box X \supset (\Box Y \supset \Box(X \wedge Y))$	from 2, 3 classically
5	$(\Box X \wedge \Box Y) \supset \Box(X \wedge Y)$	from 4, classically

We now have enough to conclude that $\Box(X \wedge Y) \equiv (\Box X \wedge \Box Y)$ is axiomatically proved.

We will eventually establish that the formulas with proofs in the axiom system we have given are exactly the **K**-valid formulas. In the next chapter we will show these are also exactly the formulas having proofs using **K** tableau rules as well.

We are less interested in giving lots of formal axiomatic proofs than we are in using features of the axiomatic treatment, often substantially abbreviated, to manipulate modal formulas in ways that preserve validity. Here is a particularly useful tool for this purpose.

Theorem 6.1.6 (Replacement) *Suppose X, X', Y , and Y' are formulas, X occurs as a subformula of Y , and Y' is like Y except that the occurrence of X has been replaced with an occurrence of X' . If $X \equiv X'$ has an axiomatic proof, so does $Y \equiv Y'$.*

This is a well-known theorem classically, and the proof of the modal version is a simple extension of the classical proof. It amounts to showing how an axiomatic proof of $X \equiv X'$ can be transformed into one of $Y \equiv Y'$. We omit the argument

here—details can be found in Chellas (1980) or Fitting (1983). Here is a typical example of an application.

Example 6.1.7 We show $\Diamond\Diamond\neg X \equiv \neg\Box\Box X$ is a theorem. First, using the Possibility axiom, and Replacement, Theorem 6.1.6, twice, it is enough to show theoremhood for $\neg\Box\neg\neg\Box\neg\neg X \equiv \neg\Box\Box X$. Now, $\neg\neg Z \equiv Z$ is a tautology for every formula Z , so by the Replacement Theorem, twice again, we need to show $\neg\Box\Box X \equiv \neg\Box\Box X$ is a theorem. But this is a tautology.

There are certain replacements that are often useful to make: $\Box\neg X \equiv \neg\Diamond X$ and $\Diamond\neg X \equiv \neg\Box X$ are two good examples. (It is easy to see that each of these is a theorem.) Repeated use of one of these quickly gives the result in the example above, for instance.

Exercises

Exercise 6.1.1 Show the following is a derived rule: conclude $\Diamond X \supset \Diamond Y$ from $X \supset Y$.

Exercise 6.1.2 In the last two examples above, abbreviated proofs are given. Give them in full.

Exercise 6.1.3 Give an axiomatic proof of each of the following.

1. $\Diamond(X \vee Y) \equiv (\Diamond X \vee \Diamond Y)$
2. $\Box(X \supset Y) \supset (\Diamond X \supset \Diamond Y)$
3. $(\Box X \vee \Box Y) \supset \Box(X \vee Y)$
4. $\Box(X \vee Y) \supset (\Box X \vee \Diamond Y)$
5. $(\Box X \wedge \Diamond Y) \supset \Diamond(X \wedge Y)$
6. $(\Diamond X \supset \Box Y) \supset \Box(X \supset Y)$

6.2 More Axiom Systems

In the previous section we gave an axiom system for **K**. Now we give versions for the other modal logics in the modal cube. It is a simple matter to do so because each logic has an axiomatization that is the same as that for **K**, but with the addition of a few extra axioms.

Definition 6.2.1 The first table below shows five modal axiom schemes and their standard names. The second shows which schemes are added to the **K** axiomatization from Definition 6.1.1 for the various logics we consider. Above the line are logics with historical names; below the line are the other logics from the modal cube, Fig. 5.1.

Scheme		Logic	Axiom Schemes Added to K
Name	Scheme		
<i>D</i>	$\Box X \supset \Diamond X$	T	<i>T</i>
<i>T</i>	$\Box X \supset X$	B	<i>T</i> + <i>B</i>
4	$\Box X \supset \Box \Box X$	S4	<i>T</i> + 4
<i>B</i>	$X \supset \Box \Diamond X$	S5	<i>T</i> + <i>B</i> + 4 or <i>T</i> + 4 + 5
5	$\Diamond X \supset \Box \Diamond X$	D	<i>D</i>
		K4	4
		K5	5
		K45	4 + 5
		KB	<i>B</i>
		KB5	<i>B</i> + 5
		KD4	<i>D</i> + 4
		KD5	<i>D</i> + 5
		KD45	<i>D</i> + 4 + 5
		KDB	<i>D</i> + <i>B</i>

The schemas above can be given in a variety of alternate forms. For instance, in axiomatic **T**, $P \supset \Diamond P$ is a theorem, by the following argument. $\Box X \supset X$ is an axiom scheme¹ of **T**, so $\Box \neg P \supset \neg P$ is an axiom. Using contraposition, $\neg \neg P \supset \neg \Box \neg P$ is a theorem, and using double negation replacement and the Possibility schema of **K**, we have theoremhood for $P \supset \Diamond P$. By a similar argument, we can show that taking $X \supset \Diamond X$ as an axiom scheme allows us to prove all instances of $\Box P \supset P$, so either $\Box X \supset X$ or $X \supset \Diamond X$ would do to axiomatize **T**. Similarly, $\Diamond \Diamond P \supset \Diamond P$ is a theorem of any system including axiom scheme 4, $\Diamond \Box P \supset \Box P$ is a theorem of any system including axiom scheme 5, and these give us substitutes for schemes 4 and 5 respectively.

In much of the literature that makes use of **S5**, an axiomatization of $T + 4 + 5$ is common. It is natural if one thinks of $\Box$ epistemically, as a knowledge operator. Here we will use the alternative axiomatization, $T + B + 4$, as our main one because it directly corresponds to the accessibility relation of a model being an equivalence relation. The following explains what we mean by this, and more as well. We look at the semantic counterparts of the various axiom schemes listed above.

Reflexive We will show this frame condition corresponds to the validity of all instances of the *T* scheme, $\Box X \supset X$. Thus it amounts to assuming that whatever is necessary is true. We will present a brief discussion of epistemic logic later, and we will make use of the scheme $K_a X \supset X$, (6.3). This is a version of the *T* scheme. In an epistemic context reflexivity on frames amounts to assuming that knowledge is factual—what an agent *a* knows is true.

¹ “nature must obey necessity,” *Julius Caesar*, William Shakespeare.

Transitive This will be shown to correspond to axiom scheme 4, $\Box X \supset \Box \Box X$.

Loosely it says that if a formula is a necessary truth it cannot be so contingently—the necessity of its truth must itself be necessary. In an epistemic setting we will consider a version of this, *positive introspection*, (6.4), $K_a X \supset K_a K_a X$, which is a fairly strong condition on an agent’s knowledge: if an agent knows something, the agent knows that it is known.

Euclidean There is a correspondence with axiom scheme 5, $\Diamond X \supset \Box \Diamond X$. It thus says that possible truth cannot be contingent—it must be necessary. The epistemic formula $\neg K_a X \supset K_a \neg K_a X$, *negative introspection*, is a variation of this. If an agent doesn’t know something, the agent knows that fact. It is actually quite improbable in general, but is commonly assumed anyway.

Symmetric This will be shown to correspond to the *B* schema, $X \supset \Box \Diamond X$. Informally, what is true must not only be possible, as with our variant $X \supset \Diamond X$ of *T*, but more strongly, a truth must be *necessarily* possible.

Serial This matches up with the axiom scheme *D*, $\Box X \supset \Diamond X$, necessity entails possibility. The equivalent version, $\Box X \supset \neg \Box \neg X$, can be thought of as a kind of consistency requirement, saying that if *X* is a necessary truth then its negation cannot be. If we are in a deontic setting, this can be read something like “if we have an obligation to bring *X* about, then we do not have an obligation to also bring the opposite about.”

Again without proof, we note that the Replacement Theorem 6.1.6 holds for the axiomatic versions of all the logics in Definition 6.2.1. We make considerable use of this fact.

In Sect. 5.4 we noted that semantically **D** was a sublogic of **T**. This can be shown for the axiomatic versions as well: it is enough to show that every theorem of axiomatic **D** is also a theorem of **T**, and to show this it is enough to show that every formula of the form $\Box X \supset \Diamond X$ is a theorem of **T**. But, in **T**, $\Box X \supset X$ is, itself, an axiom, and we saw above that $X \supset \Diamond X$ is a theorem of **T**, so $\Box X \supset \Diamond X$ follows immediately from these two, by classical logic. Similarly, axiomatic **T** is a sublogic of **S4** and **S5**, and so on.

We will see that there are several other natural axiomatizations of **S5**. For instance, **KT5** will do, that is, using the axiom system for **K**, with the addition of axioms *T* and 5. Here are proof sketches that every instance of *B* and of 4 is provable in **KT5**. We begin with *B*.

$$\begin{aligned} P &\supset \Diamond P && \text{theorem of } \mathbf{T} \\ \Diamond P &\supset \Box \Diamond P && \text{by 5} \\ P &\supset \Box \Diamond P && \text{by classical logic} \end{aligned}$$

Next we show each instance of scheme 4 is also provable. Since we have shown that instances of *B* are provable, we can make use of them in our abbreviated proofs.

$$\begin{array}{ll}
\Diamond\Box P \supset \Box P & \text{by 5} \\
\Box\Diamond\Box P \supset \Box\Box P & \text{by (derived) Regularity Rule} \\
\Box P \supset \Box\Diamond\Box P & \text{by } B \\
\Box P \supset \Box\Box P & \text{by classical logic}
\end{array}$$

We will see more about this later.

Exercises

Exercise 6.2.1 In the axiom system **K** plus axiom schemes *T*, 4 and *B*, show each instance of axiom scheme 5 is a theorem.

Exercise 6.2.2 Prove $(\Box X \wedge \Box Y) \supset \Box(\Box X \wedge \Box Y)$ in the **K4** axiom system.

Exercise 6.2.3 Prove $\Diamond(P \supset \Box P)$ in the **T** axiom system.

6.3 Logical Consequence, Axiomatically

We have seen in Sect. 5.6 that logical consequence is more complex for modal logic than it is for classical logic. Local assumptions and global assumptions play quite different roles. This difference is somewhat clarified by the axiomatic conditions appropriate for the two kinds of assumptions. But before going into the details, there is an important point to bring up. As we treat things, *axioms* are given via axiom *schemes*. That is, all formulas of a particular form count as axioms. But when we talk about consequence, local or global, it is *particular formulas* whose consequences we consider, and not formula schemes. For instance, we might want to consider what the consequences (local or global) of assuming *P* are, where *P* is a particular propositional letter. If *P* were meant as a scheme, we would be considering the consequences of *all* formulas, which would not be very interesting. It is the propositional letter itself we would want. With this understood, we move on to the representation of local and global consequence axiomatically.

Suppose a particular formula *X* is true at every world of some model $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$. If Γ is any world of this model, *X* will be true at every world accessible from Γ (since *X* is true at every world), so $\Box X$ will be true at Γ . Since Γ was arbitrary, $\Box X$ must be true at every world of the model. Now global assumptions, semantically, are true at every world of a model, so we have just argued that if *X* is a global assumption, the Rule of Necessitation can be applied to *X*—we can assume $\Box X$ is also the case.

On the other hand, if *X* is known to be true only at a particular world Γ of some model, we have no reason to suppose $\Box X$ is also true there. It is easy to construct

examples for which this is not so. Then if X is a local assumption, the Rule of Necessitation cannot be assumed to apply to it.

Formally, the distinction between global and local assumptions in axiomatic derivations comes down to the applicability or not of the Rule of Necessitation.

Definition 6.3.1 (Axiomatic Derivation) Let $\mathbf{L}$ be one of the modal logics from Definition 6.2.1. By an axiomatic *derivation* in $\mathbf{L}$ using members of S as global assumptions and members of U as local assumptions, we mean a sequence of formulas meeting the following conditions.

1. The sequence is divided into two separate parts, a *global* part and a *local* part, with the global part coming first.
2. In the global part, each formula is either an axiom of $\mathbf{L}$, a member of S , or follows from earlier lines by *Modus Ponens* or Necessitation.
3. In the local part, each formula is either an axiom of $\mathbf{L}$, a member of U , or follows from earlier lines by *Modus Ponens* (but not by Necessitation).

If X is the last formula in the sequence, we say the sequence is a derivation of X .

Thus the effect is that the Necessitation Rule can be used with global, but not with local assumptions. Here is an example of a derivation.

Example 6.3.2 We show that $\Box P$ has a derivation in the logic $\mathbf{K}$ from $\{\Box P \supset P\}$ as global and $\{\Box\Box P\}$ as local assumptions. See Exercise 5.6.1 for some background to this example. In the following, lines 1 and 2 are in the global part, lines 5 and 6 are in the local part, and lines 3 and 4 can be counted either way—it doesn't matter.

1 $\Box P \supset P$	global assumption
2 $\Box(\Box P \supset P)$	Necessitation, on 1
3 $\Box(\Box P \supset P) \supset (\Box\Box P \supset \Box P)$	$\mathbf{K}$ axiom
4 $\Box\Box P \supset \Box P$	<i>Modus Ponens</i> on 2, 3
5 $\Box\Box P$	local assumption
6 $\Box P$	<i>Modus Ponens</i> on 4, 5

Exercises

Exercise 6.3.1 First, give an axiomatic derivation in $\mathbf{K}$ of $\Box\Box P \supset Q$ using $\{\Box P \supset P\}$ as global and $\{P \supset Q\}$ as local assumptions. That is, we have an axiomatic version of $\{\Box P \supset P\} \models_{\mathbf{K}} \{P \supset Q\} \rightarrow \Box\Box P \supset Q$. Then give a model to show we do not have $\{\Box P \supset P\} \models_{\mathbf{K}} \{P \supset Q\} \rightarrow \Box\Box P \supset \Box Q$.

Exercise 6.3.2 From Definition 6.2.1, T is the axiom scheme $\Box X \supset X$. Suppose we let $\llbracket T \rrbracket$ be the set of all instances of this scheme. Show that for each modal formula Z , Z is provable in $\mathbf{T}$ if and only if Z has a derivation in axiomatic $\mathbf{K}$ with $\llbracket T \rrbracket$ as global assumptions and no local assumptions. (Similar things can be

done with other modal logics from Definition 6.2.1. This essentially shows that, axiomatically, global consequence reduces everything to **K**.)

6.4 Axiom Systems Work

In Chap. 2 we proved soundness and completeness for a *classical* propositional axiom system. Now we extend that to cover the modal systems introduced here. Once this is done, we will know that for **L** being any of the modal logics semantically characterized in Definition 5.4.3 and axiomatically characterized in Definition 6.2.1, a formula X is **L**-valid if and only if it has an axiomatic **L** proof. The machinery is much more general than just the modal cube, but that is beyond what we need in this book. Our proofs can be extended to take logical consequence into account, and we leave this as exercises.

6.4.1 Soundness

As we saw for classical logic, soundness for an axiomatic system is easy to prove. We simply show each of the axioms is valid, and the rules of inference produce valid formulas from valid formulas. Then it follows that every line of a proof must be a valid formula, hence the last line—the formula being proved—is also valid.

We begin with the rules of inference. And of these, *Modus Ponens* is easy. Suppose $P \supset Q$ is true at a world of some model. Then either P is not true there, or Q is. So if P is also true at that world, Q must be. It follows that if P and $P \supset Q$ are both valid in a model (that is, if both are true at every world of the model), Q must also be valid in that model.

The Rule of Necessitation is almost as easy, but we cannot work one world at a time as we did with *Modus Ponens*. Suppose P is valid in a modal model. We show $\Box P$ is also valid in that model. Take Γ to be an arbitrary world of the model; we show $\Box P$ is true at Γ . To do this, let Δ be any world that is accessible from Γ ; it is enough to show P is true at Δ . But this is the case since P is valid in the model, and hence true at every possible world.

We have shown that both rules of inference turn formulas that are valid in a model into other formulas that are also valid in that model. Now we look at the axioms for our logics, and we begin with those for **K**.

At each world of a modal model the propositional connectives have their usual classical behavior. It follows immediately that every tautology is true at every possible world of every model—hence tautologies are valid in every model. Exercise 5.2.2 asked you to show what is needed for the Possibility axiom scheme. Likewise, in Exercise 5.3.4 you were asked to show the validity, in every model, of formulas of the form $\Box(X \supset Y) \supset (\Box X \supset \Box Y)$. Thus all the **K** axioms are valid in all modal models.

We have now shown enough to establish the soundness of the **K** axiom system with respect to the **K** semantics. Extending soundness to the remaining modal logics is a simple matter. The following provides everything we need.

Theorem 6.4.1 *Each of the axiom schemes listed in the first column below is valid in every frame that meets the corresponding condition listed in the second column.*

<i>Scheme</i>	<i>Condition</i>	
<i>D</i>	<i>serial</i>	
<i>T</i>	<i>reflexive</i>	
<i>B</i>	<i>symmetric</i>	(6.1)
<i>4</i>	<i>transitive</i>	
<i>5</i>	<i>Euclidean</i>	

Proof Exercise 5.4.1 provides what is needed for axiom scheme *D*. Example 5.3.7 shows we have the 4 case, connecting with transitivity. Exercise 5.4.3 asks you to take care of the rest.

Corollary 6.4.2 (Axiom System Soundness) *If X has a proof using the axiom system resulting from adding any of the axiom schemes D , T , B , 4 , and 5 to axiomatic **K**, then X is valid in the family of frames meeting the corresponding conditions from Theorem 6.1.*

6.4.2 Completeness

The first completeness proofs for modal axiom systems with respect to possible world models derived axiomatic completeness from the completeness of tableau systems, Kripke (1959, 1963a,b). The completeness of tableau systems was established directly. Indeed, Negri (2009) makes a case that this is the best way to do it. But today, modal axiomatic completeness is almost universally proved by an argument based on Lindenbaum’s Lemma, which was covered in Sect. 2.4. This is so much the case that people who are familiar with Kripke’s work by hearsay, and not from his actual writings, often believe that the Lindenbaum approach is due to him.

The actual origin of the completeness proof presented here is a bit fuzzy. The following is information from Scott (2022). The now-common style of proving modal completeness came out of California (Berkeley, Stanford, UCLA, and probably more). Dana Scott and John Lemmon collaborated until Lemmon’s early death in 1966. This led to what are commonly called the *Lemmon Notes*, which had a substantial informal circulation. According to Scott, “Dov Gabbay visited Stanford in the mid-1960s, and at various times Krister Segerberg and Brian Chellas were my students there. Among all these people there was a lot of discussion of modal logic

and completeness proofs.” The Lindenbaum style of proving modal completeness somehow came out of all this.

The Lemmon Notes were not published until Lemmon and Scott (1977). The first actual publication of the proof method was in Makinson (1966), and was based on part of his dissertation from 1965. But Makinson’s article appeared in what was, at the time, an East German journal and was not as widely read as, say, *Journal of Symbolic Logic*. As it happened, at around the same time Kaplan (1966) described the Lindenbaum style argument in his influential review of Kripke’s modal work. Many, including one of the authors of the present book, learned about it from this review. It is a curious route for a standard to be made available to the general public.

As it happens, the Lindenbaum style argument applies to tableau proof procedures, as well as to axiom systems. We saw this in the subsection entitled *Tableau Completeness, Non-Constructively* in Sect. 3.4. We will see it again when it comes to proving completeness for modal tableaux, in Sect. 7.6. In fact, a version of the argument can be used to prove completeness of the tableau systems that Kripke originally introduced, without the complexities of describing systematic tableau constructions that he had to resort to, and which Kaplan complained about.

In Chap. 2 completeness was proved for axiomatically presented classical propositional logic using maximally consistent sets of formulas. For a modal logic the language is larger, containing necessity and possibility symbols, there are more axioms, and most importantly there is an additional rule of derivation, necessitation. But the basic results from the earlier chapter still hold, and still play a fundamental role. The place to begin is, exactly what should “consistent” mean now?

In our axiomatic presentation of classical propositional logic, inconsistency and consistency were characterized in Definition 2.4.1: an inconsistent set has every formula as a consequence. Later we saw in Sect. 2.9, specifically in Proposition 2.9.2, that we could replace entailing every formula with just entailing a contradiction, with any formula of the form $P \wedge \neg P$ serving as the contradiction. We want to apply this idea modally, but we first need to say what “entail” means, since now we have two versions of consequence, local and global. In addition, while axiomatic classical propositional logic only had one rule of inference, *Modus Ponens*, we now have a second rule, *Necessitation*.

Classically, the fact that *Modus Ponens* was the only rule of inference was made use of in proving the Deduction Theorem, 2.5.1. In that proof a *Modus Ponens* application, from Z_i and $Z_i \supset Z_k$ derive Z_k , was ‘internalized’ using the axiom $(X \supset (Z_i \supset Z_k)) \supset ((X \supset Z_i) \supset (X \supset Z_k))$, something that is a tautology. As we said, we now also have *Necessitation* as a rule of inference. In a proof, if we use it to conclude $\Box Z$ from Z , the needed ‘internalization’ would be $(X \supset Z) \supset (X \supset \Box Z)$, but this is not a modal validity of any of the logics we have seen.

Local and global modal consequence were defined semantically in Sect. 5.6 and axiomatically in Sect. 6.3. The semantic and axiomatic versions will turn out to be equivalent but technically, we don’t know that yet. Still, suppose we assume they are and, working under that assumption, see if we can come up with a plausible and useful definition of axiomatic consistency.

Axiomatically, the problematic *Necessitation* rule can be used when working with global premises, but not with local ones. So as a first pass, to avoid problems with Necessitation we could define modal inconsistency for a set U of formulas as: U has $P \wedge \neg P$ as a *local* consequence. It is not hard to check that the proof of the Local Deduction Theorem, 5.6.6, still goes through axiomatically. That is, it can be shown that Y has a derivation from $\emptyset$ as global and $U \cup \{X\}$ as local assumptions if and only if $X \supset Y$ has a derivation from $\emptyset$ as global and U as local assumptions. Further, all derivations are finite, so only a finite subset of U will be needed. Using the modal version of the local Deduction Theorem multiple times then gives us the following: a set U has $P \wedge \neg P$ as a local consequence if and only if U has a finite subset $\{A_1, \dots, A_n\}$ that has $P \wedge \neg P$ as a local consequence, if and only if U has a finite subset $\{A_1, \dots, A_n\}$ such that $(A_1 \supset (A_2 \supset \dots \supset (A_n \supset (P \wedge \neg P)) \dots))$ is locally derivable from the empty set of premises. Or, more simply, U has $P \wedge \neg P$ as a local consequence if and only if U has a finite subset $\{A_1, \dots, A_n\}$ such that $(A_1 \wedge \dots \wedge A_n) \supset (P \wedge \neg P)$ is provable.

This rather informal discussion leads us to the following for the formal definition of consistency we will use.

Definition 6.4.3 (Axiomatic Modal Consistency) Let $\mathbf{L}$ be one of the modal logics from Definition 6.2.1. A set S of formulas is *inconsistent* if there is a finite subset $\{A_1, \dots, A_n\}$ of S such that $(A_1 \wedge \dots \wedge A_n) \supset (P \wedge \neg P)$ is provable in $\mathbf{L}$. S is *consistent* if it is not inconsistent, and S is *maximally consistent* if it is consistent but no proper extension is consistent. To make clear what modal logic we are using, we may say a set is *$\mathbf{L}$ consistent*, *$\mathbf{L}$ inconsistent*, and so on.

Properly speaking, we should not be writing $(A_1 \wedge \dots \wedge A_n) \supset (P \wedge \neg P)$, since conjunction is binary. It can be shown that all legal ways of inserting parentheses into $(A_1 \wedge \dots \wedge A_n)$ are provably equivalent, so we can be conveniently informal here. Such simplifications will be made throughout, without further comment.

Lindenbaum's Lemma 2.4.7 still holds and with the same proof. The enlargement of the language with the necessity and possibility symbols, and the addition of modal axioms make no difference. We restate it here for convenience.

Theorem 6.4.4 (Lindenbaum's Lemma) Let S be any set that is $\mathbf{L}$ consistent. Then $S \subseteq S_\infty$ for some maximally $\mathbf{L}$ consistent set S_∞ .

Here is the key new feature in the modal completeness proof, whose somewhat murky origins we discussed earlier.

Proposition 6.4.5 If the set $\{\neg \Box B, \Box A_1, \Box A_2, \dots\}$ is $\mathbf{L}$ -consistent, so is the set $\{\neg B, A_1, A_2, \dots\}$.

Proof We show the contrapositive. Suppose $\{\neg B, A_1, A_2, A_3, \dots\}$ is not $\mathbf{L}$ -consistent; we show that neither is $\{\neg \Box B, \Box A_1, \Box A_2, \dots\}$. By assumption, $\{\neg B, A_1, A_2, A_3, \dots\}$ has a finite subset that entails a contradiction. If a finite set is not $\mathbf{L}$ -consistent, neither is any extension of it, so we can assume we have a finite subset that consists of $\neg B$, and all of $A_1, A_2, \dots, A_n$, up to some n , and that implies a contradiction. Now we proceed as follows.

$(\neg B \wedge A_1 \wedge \dots \wedge A_n) \supset (P \wedge \neg P)$	assumption
$(A_1 \wedge \dots \wedge A_n) \supset (\neg B \supset (P \wedge \neg P))$	exportation
$(A_1 \wedge \dots \wedge A_n) \supset B$	since $(\neg B \supset (P \wedge \neg P)) \equiv B$
$\Box(A_1 \wedge \dots \wedge A_n) \supset \Box B$	Regularity Rule
$(\Box A_1 \wedge \dots \wedge \Box A_n) \supset \Box B$	Example 6.1.5
$(\Box A_1 \wedge \dots \wedge \Box A_n) \supset (\neg \Box B \supset (P \wedge \neg P))$	since $(\neg \Box B \supset (P \wedge \neg P)) \equiv \Box B$
$(\neg \Box B \wedge \Box A_1 \wedge \dots \wedge \Box A_n) \supset (P \wedge \neg P)$	importation

So follows that $\{\neg \Box B, \Box A_1, \Box A_2, \Box A_3, \dots\}$ has a finite subset that implies a contradiction, so it is not **L**-consistent.

We have the preliminary work out of the way. Next we say how to construct what is called the *canonical model* for the logic **L**. Recall that in the completeness proof for classical propositional logic in Sect. 2.8, we made use of a maximally consistent set to create a boolean valuation. Now we make use of *all* maximally consistent sets to create a modal model.

Definition 6.4.6 (Canonical Model for Axiomatic Modal Logic **L)** Using modal logic **L**, one of those from the modal cube in Definition 6.2.1, we create a modal model $\mathcal{M}$ as follows. The set of possible worlds $\mathcal{G}$ is the set of *all* maximally **L**-consistent sets of formulas. If Γ and Δ are in $\mathcal{G}$, we set $\Gamma \mathcal{R} \Delta$ provided that, for each formula in Γ of the form $\Box Z$, the corresponding formula Z is in Δ . And finally, for each propositional letter P and each $\Gamma \in \mathcal{G}$, we set $\Gamma \Vdash P$ just in case $P \in \Gamma$ (recall, members of $\mathcal{G}$ are sets of formulas, so this condition makes sense). We have now completely defined a model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$. It is called the *canonical model* for **L**.

In the canonical model possible worlds are actually sets of formulas, and thus have a kind of dual personality: formulas *belong* to them, and formulas are *true* at them. Here is the fundamental result connecting these two aspects of possible worlds in the canonical model.

Proposition 6.4.7 (Truth Lemma) *Let $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ be the canonical model for **L**. For every formula Z and for every $\Gamma \in \mathcal{G}$,*

$$\Gamma \Vdash Z \text{ if and only if } Z \in \Gamma$$

Proof The argument is by Complete Induction on the degree of the formula Z , Definition 5.1.2. Let $\Gamma \in \mathcal{G}$. There are several cases.

Ground Case If P is a propositional variable, $P \in \Gamma$ if and only if $\Gamma \Vdash P$, by definition of the model.

Binary Cases The cases for $\wedge$, $\vee$, and $\supset$ are by Propositions 2.6.1, 2.6.2, and 2.6.4.

Unary Cases The negation case is by Proposition 2.6.3.

Modal Cases We first show the result for $\Box$. Suppose $Z = \Box W$ and the result is known for W , which is a formula of lower degree. We have a two-part argument.

Right To Left Assume that $Z \in \Gamma$, that is, $\Box W \in \Gamma$. Let Δ be an arbitrary member of $\mathcal{G}$ such that $\Gamma \mathcal{R} \Delta$. By our definition of $\mathcal{R}$, since $\Box W$ is in Γ , W is in Δ . By the induction hypothesis, $\Delta \Vdash W$. Since Δ was arbitrary, W is true at every member of $\mathcal{G}$ accessible from Γ , hence $\Gamma \Vdash \Box W$, that is, $\Gamma \Vdash Z$.

Left To Right Assume that $Z \notin \Gamma$, that is, $\Box W \notin \Gamma$, and so we must have $\neg \Box W \in \Gamma$ by maximal consistency. Let $\{\Box Y_1, \Box Y_2, \Box Y_3, \dots\}$ be the set of all members of Γ that begin with $\Box$. Then $\{\neg \Box W, \Box Y_1, \Box Y_2, \Box Y_3, \dots\}$ is a subset of Γ , and so is $\mathbf{L}$ -consistent. By Proposition 6.4.5, $\{\neg W, Y_1, Y_2, Y_3, \dots\}$ is also $\mathbf{L}$ -consistent. Using Theorem 6.4.4 it extends to a maximal $\mathbf{L}$ -consistent set Δ . Then $\Delta \in \mathcal{G}$. Also if $\Box Y$ is in Γ then Y is in Δ by construction, and hence $\Gamma \mathcal{R} \Delta$. Finally, $\neg W \in \Delta$, so $W \notin \Delta$, and by the induction hypothesis, $\Delta \not\Vdash W$. But then it follows that $\Gamma \not\Vdash \Box W$, so $\Gamma \not\Vdash Z$.

Finally we have the modal case for $\Diamond$, which is easy since we can make use of the $\Box$ case that was just proved. By the Possibility Axiom, $\Diamond X \equiv \neg \Box \neg X$. Then using the maximal consistency of Γ , $\Diamond X \in \Gamma$ iff $\neg \Box \neg X \in \Gamma$ iff $\Box \neg X \notin \Gamma$ iff $\Gamma \not\Vdash \Box \neg X$ iff $\Gamma \Vdash \neg \Box \neg X$ iff $\Gamma \Vdash \Diamond X$.

This completes the proof of the Truth Lemma.

We have all the preliminary work out of the way. Here is the main event.

Theorem 6.4.8 (Axiom System Completeness) *Let modal logic $\mathbf{L}$ be one of those specified axiomatically in Definition 6.2.1 and semantically in the modal cube of Fig. 5.1. If X is $\mathbf{L}$ -valid then X has a proof using the $\mathbf{L}$ axiom system.*

Proof We show the contrapositive: suppose X is a formula that has no axiomatic $\mathbf{L}$ proof—we show it is not $\mathbf{L}$ -valid.

Since X has no proof, the set $\{\neg X\}$ is $\mathbf{L}$ -consistent, because if it were not, $\neg X \supset (P \wedge \neg P)$ would be provable, and this is equivalent to X even in classical logic. Using Theorem 6.4.4, extend $\{\neg X\}$ to a maximally $\mathbf{L}$ -consistent set, call it Γ_0 .

Construct the canonical model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ for $\mathbf{L}$, Definition 6.4.6. Since Γ_0 is maximally $\mathbf{L}$ -consistent, it is a member of $\mathcal{G}$. But also, $\neg X \in \Gamma_0$, since Γ_0 extends $\{\neg X\}$. Then $X \notin \Gamma_0$, and by the Truth Lemma 6.4.7, $\Gamma_0 \not\Vdash X$. Thus there is a world, Γ_0 , of a modal model $\mathcal{M}$, the canonical model for $\mathbf{L}$, at which X is false.

What remains is to show that the canonical model $\mathcal{M}$ for $\mathbf{L}$ actually meets the conditions for being an $\mathbf{L}$ model. If $\mathbf{L}$ happens to be $\mathbf{K}$ we are done right now, since there are no special conditions a $\mathbf{K}$ model must meet. For the rest, what we must show is: if D is one of the $\mathbf{L}$ axiom schemes then the canonical $\mathbf{L}$ model satisfies seriality; if T is one of the axiom schemes of $\mathbf{L}$ then the canonical $\mathbf{L}$ model is transitive, and so on through the five cases shown in Theorem 6.1. We show one case and leave the rest to you in Exercise 6.4.3. Suppose 4 is an axiom scheme of $\mathbf{L}$, $\Box X \supset \Box \Box X$; we show the canonical $\mathbf{L}$ model is transitive.

Let $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ be the canonical $\mathbf{L}$ model, where $\Box X \supset \Box \Box X$ is an $\mathbf{L}$ axiom scheme. Suppose Γ, Δ, Ω are in $\mathcal{G}$, and $\Gamma \mathcal{R} \Delta$ and $\Delta \mathcal{R} \Omega$. We must prove that $\Gamma \mathcal{R} \Omega$. For this, we assume $\Box X \in \Gamma$ and we show $X \in \Omega$, for an arbitrary formula X . From the way we defined $\mathcal{R}$, this will be enough to get what we want. The

argument is simple. We assumed $\Box X \in \Gamma$, also $\Box X \supset \Box\Box X$ is an axiom so it is in Γ by Exercise 6.4.2. Maximally consistent sets are closed under *Modus Ponens*, by the same Exercise. Then $\Box\Box X \in \Gamma$. Since $\Gamma \mathcal{R} \Delta$, by definition $\Box X \in \Delta$. Then, since $\Delta \mathcal{R} \Omega$, we have $X \in \Omega$, which is what we wanted.

We have a few concluding remarks. First, an axiomatically formulated modal logic **L** is called *canonical* if the canonical model for **L** meets the semantic conditions appropriate for **L**. In these terms, a central part of the completeness argument just given is: every logic in the modal cube is canonical. This is not something that always happens—there are modal logics that are sound and complete with respect to a class of frames, but are not canonical. It is a complex subject, taken as a whole. The book Blackburn et al. (2001) gives a thorough overview for those interested, but it is more than we will go into here.

Our second remark has to do with the modal cube. We formulated it semantically in Fig. 5.1. Loosely, it was to represent all combinations of reflexivity, symmetry, transitivity, seriality and Euclideaness. But this would be 32 items, and the cube has 15. This was because sometimes more than one combination of conditions determined the same class of frames. For instance, **S5** was determined in 9 different ways. We picked **KTb4** as our representative, reflexivity, symmetry, and transitivity. A very common equivalent choice in the literature is **KT45**. As a matter of fact, even though the two are equivalent, weakening **T** to **D** does not do the same thing with the two versions. **KDb4** is simply another equivalent to **KTb4**, but **KD45** is a different class of frames than **KT45**, and has been found to have independent interest in some applications.

When we use, say, **KTb4** and **KT45** semantically, they are the same class of frames. But when used axiomatically, they are different systems. As it is stated, Theorem 6.4.8 obscures this distinction since it refers to the semantic cube where axiomatic distinctions are blurred, and only one name is associated with each node. However, the completeness results still apply (as do the soundness results). Our proof actually shows completeness of **KTb4** with respect to the class of reflexive, symmetric, and transitive frames. And it shows completeness of **KT45** with respect to reflexive, transitive, and Euclidean frames. Exercise 5.5.1 asked you to show these are the same classes of frames.

Exercises

Exercise 6.4.1 Show every subset of an **L**-consistent set is **L**-consistent.

Exercise 6.4.2 Assume that S is maximally **L**-consistent.

1. Show that if $S \cup \{Z\}$ is **L**-consistent, then $Z \in S$.
2. Show that if X is a theorem of the **L** axiom system, then $X \in S$.
3. Show that if $X \in S$ and $(X \supset Y) \in S$ then $Y \in S$.

Exercise 6.4.3 Show that for any **L** from the modal cube:

1. if $\mathbf{L}$ has D as an axiom scheme then the canonical model satisfies seriality;
2. if $\mathbf{L}$ has T as an axiom scheme then the canonical model satisfies reflexivity;
3. if $\mathbf{L}$ has B as an axiom scheme then the canonical model satisfies symmetry;
4. if $\mathbf{L}$ has 5 as an axiom scheme then the canonical model satisfies the Euclidean condition.

6.5 Informal Notes

6.5.1 Gödel's Intuitionistic Logic Interpretation

The axiomatization we used for the modal system $\mathbf{S4}$ can be traced back to Kurt Gödel's axiomatic characterization of the informal notion of provability, Gödel (1933). Using Bp to designate "It is provable that p " [*p ist beweisbar*], he introduced a system which includes all the axiom schemes and rules of traditional propositional classical logic, and also

$$B(X \supset Y) \supset (BX \supset BY) \quad (\mathbf{K})$$

$$BX \supset X \quad (\mathbf{T})$$

$$BX \supset BBX \quad (4)$$

plus the additional rule of inference: from X infer BX .

Gödel's B corresponds to our $\Box$, and this explains our using the conventional modal labels for his axioms. Put in terms of *provable*, however, their appropriateness becomes obvious. The first of the listed axioms, the K axiom, is an internalization of *Modus Ponens*: if a conditional is provable, then if the antecedent is provable, then the consequent is provable. The second axiom, the T axiom, says that if a formula is provable, it is true. This is surely a desirable property for any proof system. The last, the 4 axiom, says that if a formula is provable, then the fact that it is provable is itself provable. This is assured by a familiar constraint on formal systems: we must be able to establish, in a finite number of steps, that a given proof is indeed a proof.

It is important to recognize, as Gödel himself did, that while the T axiom here holds for our informal notion of provable, it does not hold for the specific case of provability in Peano Arithmetic. Suppose we let X be a contradiction, $P \wedge \neg P$. Then an instance of T is $B(P \wedge \neg P) \supset (P \wedge \neg P)$. By classical logic this is equivalent to $\neg B(P \wedge \neg P)$, that is, non-provability of a contradiction, in other words, consistency. But Gödel's Second Incompleteness Theorem says that using the machinery of Peano arithmetic, we cannot prove the consistency of Peano Arithmetic. We cannot have this, and so T does not apply if provable means provable in Peano arithmetic. Loosely, we are not assured that a provable formula must be true. Other modal

axiom systems have been offered that do capture the notion of provability in Peano Arithmetic, see Boolos (1979, 1993).

In Gödel's system, the connectives are all understood classically, and proof is clearly differentiated from truth. In Intuitionistic logic, however, proof enjoys the role truth plays classically. So, whereas classically, to assert a given statement is equivalent to asserting that the statement is true, intuitionistically, to assert a given statement is equivalent to asserting that it is proved. This explains Gödel's reason for axiomatizing the abstract notion of a proof. He introduced a translation scheme (actually, more than one) to translate the statements of Arend Heyting's Intuitionistic logic, Heyting (1956), so that theorems of Intuitionistic logic would be translated into theorems of the provability logic just described. Start with an Intuitionistic formula and place a B in front of each subformula. Then the original formula is provable in Intuitionistic Logic iff the translated formula is provable in Gödel's provability logic, see Mints (2012).

Signature ideas of Intuitionism are the rejection of traditional *Reductio* proofs² and the rejection of the Law of Excluded Middle.³ As to *Reductio*, if a contradiction follows from the assumption A , one can infer its denial $\neg A$; this is not a problem for an Intuitionist. Likewise if a contradiction follows from the assumption $\neg A$, one can infer *its* denial, $\neg\neg A$. But from $\neg\neg A$ one cannot infer A , as one does classically. Double negation elimination ($\neg\neg A \supset A$) is rejected. That, in turn, requires the rejection of excluded middle ($\neg A \vee A$), see Exercise 6.6.2. In addition, Intuitionism holds that to establish the existence of some mathematical object, $(\exists x)Fx$, it is insufficient to show that the denial of its existence leads to a contradiction, one must actually produce the object, known as a 'witness', Fa , using as one's building blocks objects whose existence has already been established.

Kripke's semantics for **S4** can be thought of as capturing this Intuitionistic understanding. Instead of a system of possible worlds, we envisage a monotonic system of states of knowledge ordered by a relation that is reflexive and transitive. A proposition P holds at a state iff that proposition has been proven (or justified or verified) either in that state or in any of its predecessor states. In this semantics, a proposition's being proved true amounts to its being true at some stage in the investigation, and of course it will remain true at every subsequent stage. Indeed, Kripke (1965) provides a direct semantics for intuitionistic logic, based on the idea of states of knowledge, which increase as we learn and prove things. It will not be explored here, however.

² To prove A , assume $\neg A$ and derive a contradiction.

³ $A \vee \neg A$

6.5.2 Epistemic Logic

As we have already noted, we can interpret the usual modal symbols to create a logic of knowledge. Hintikka (1962) introduces the following readings for $\Box$ and $\Diamond$ respectively:

- $\mathcal{K}_a P$: Agent a knows that P
 $\mathcal{P}_a P$: It is possible, for all that agent a knows, that P

This is a multi-modal logic: each subscript identifies an individual. There is a similar vocabulary for belief:

- $\mathcal{B}_a P$: Agent a believes that P
 $\mathcal{C}_a P$: It is compatible with everything agent a believes that P

In each case, we have the usual interdefinability of the modal operators:

$$\mathcal{K}_a P \equiv \neg \mathcal{P}_a \neg P$$

$$\mathcal{B}_a P \equiv \neg \mathcal{C}_a \neg P$$

The theorems of Hintikka (1962) include all tautologies plus

$$\mathcal{K}_a(P \supset Q) \supset (\mathcal{K}_a P \supset \mathcal{K}_a Q) \quad (6.2)$$

$$\mathcal{K}_a P \supset P \quad (6.3)$$

$$\mathcal{K}_a P \supset \mathcal{K}_a \mathcal{K}_a P \quad (6.4)$$

and the two rules of inference, *Modus Ponens* (Y follows from X and $X \supset Y$) and *Necessitation* ($\mathcal{K}_a X$ follows from X). Formula (6.2) is the **K** axiom, for each agent. Likewise (6.3) is our **T** axiom; and (6.4) is our **4** axiom. With **T** and **4** on top of **K**, we again have an **S4** system.

$\mathcal{K}_a P$ is perhaps better read as *It follows from what a knows that P* , or perhaps as *a is entitled to know that P* . For, in this system, one knows all the logical consequences of anything one knows: if $P \supset Q$ is logically true, and if $\mathcal{K}_a P$, then $\mathcal{K}_a Q$. In this logic, then, the knower is treated as *logically omniscient*, one knows the consequences of what one knows. This feature of the system is made even more puzzling because of (6.4), which is called the *Principle of Positive Introspection*: one cannot know something and yet fail to know that one knows it. Suppose, for example, that Goldbach's Conjecture is provable (although we have no proof as yet). Then, anyone who knows that the Peano Axioms are true—most likely, any reader of this book—knows, by logical omniscience, that Goldbach's Conjecture is true, and knows, by (6.4), that they know that Goldbach's Conjecture is true. Yet it is unlikely that any reader of this book is in possession of a proof of Goldbach's conjecture, so it is unlikely that any reader of this book knows that Goldbach's Conjecture is true, and even more unlikely that any reader of this book knows that they know that Goldbach's Conjecture is true. Formula (6.4) is justified, despite its

title, not because knowing that P carries with it a certain mental feeling which one knows one has by introspection. Hintikka is working with the view that knowledge is justified, true belief.⁴ So (6.4) is justified because, if one knows that P , one must be justified in believing that P , and one cannot be justified without being aware of being justified. It is not just that if one knows that P , there are reasons justifying P ; rather, if one knows that P , these justifying reasons must be *your* reasons, and so you cannot help but be aware of them. Logical Omniscience simply tells us that if Q follows from P on logical grounds, every knower is aware of these grounds. This is a very strong condition on knowledge, and it is why we prefer reading $\mathcal{K}_a P$ as *a is entitled to know that P* , rather than as *a knows that P* .

Of course, there are some constraints on knowledge. It is interesting, in this regard, to note that by contrast with (6.4), the *Principle of Negative Introspection*,

$$\neg \mathcal{K}_a P \supset \mathcal{K}_a \neg \mathcal{K}_a P \quad (6.5)$$

is *not* a theorem of this system. (Adding it turns the above into an **S5** logic.) Again, this has nothing to do with introspection. The reasons are a bit more mundane: you might not know that P and yet *believe* that you do know it. Equation (6.5), however, requires that if you don't know that P , then you *know*, and so *believe*, you don't know it. The following also fails to be a theorem:

$$P \supset \mathcal{K}_a \mathcal{P}_a P \quad (6.6)$$

And this seems appropriate. For, a statement P might be true and yet one need not know that it is compatible with everything one believes. The following all do hold in Hintikka's system:

$$\mathcal{K}_a \mathcal{K}_b P \supset \mathcal{K}_a P \quad (6.7)$$

$$\mathcal{K}_a P \supset \mathcal{P}_a P \quad (6.8)$$

$$\mathcal{K}_a P \equiv \mathcal{K}_a \mathcal{K}_a P \quad (6.9)$$

(6.7) expresses the transmissibility of *knowledge*.⁵ Equation (6.8) essentially tells us that every world has an alternative, which we know from the reflexivity expressed in axiom (6.3). It can also be shown that $\mathcal{K}_a P \supset \mathcal{K}_a Q$ follows from $P \supset Q$. This is a form of logical omniscience: one knows all the logical consequences of anything one knows. As we noted above, it is a controversial feature of the system. In some sense, we are talking about an idealized version of knowledge, and not knowledge as people generally use the term.

⁴ "Believe it, for you know it." *Merry Wives of Windsor*, William Shakespeare.

⁵ We do not have the corresponding principle for *belief*, i.e. $\mathcal{B}_a \mathcal{B}_b P \supset \mathcal{B}_a P$.

A logic of belief is more problematic than a logic of knowledge because we are not entirely consistent in our belief sets. In any event, Hintikka (1962) suggests that

$$\mathcal{B}_a P \supset \mathcal{B}_a \mathcal{B}_a P \quad (6.10)$$

is true. In defense of (6.10), Hintikka argues that the two formulas, $\mathcal{B}_a P$ and $\mathcal{B}_a(P \supset \neg \mathcal{B}_a P)$ are inconsistent. On the other hand, he suggests that none of the following are true:

$$\mathcal{B}_a \mathcal{B}_a P \supset \mathcal{B}_a P$$

$$\mathcal{B}_a \mathcal{B}_b P \supset \mathcal{B}_a P$$

$$\mathcal{B}_a P \supset \mathcal{K}_a \mathcal{B}_a P$$

The semantics of Hintikka (1962) is in terms of what he calls *model sets*, and his proof procedure is a precursor of what has since become known as *tableau rules*. On the usual semantics for classical logic, a model consists of an assignment of a truth value to every propositional symbol and an interpretation of the logical connectives as boolean functions on these truth values, so that each complex sentence of the language is assured a unique truth value. Hintikka, by contrast, takes a model to be a *model set*, that is, a collection of sentences (deemed to be true) which might be thought of as a partial description of the world. A model set μ is governed by the following rules, which you should recognize as embodying the conditions for what we are calling a *Hintikka set*:

- (C. $\neg$) If $P \in \mu$ then not $\neg P \in \mu$
- (C. $\wedge$) If $P \wedge Q \in \mu$ then $P \in \mu$ and $Q \in \mu$
- (C. $\vee$) If $P \vee Q \in \mu$ then $P \in \mu$ or $Q \in \mu$ (or both)
- (C. $\neg\neg$) If $\neg\neg P \in \mu$ then $P \in \mu$
- (C. $\neg\wedge$) If $\neg(P \wedge Q) \in \mu$ then $\neg P \in \mu$ or $\neg Q \in \mu$ (or both)
- (C. $\neg\vee$) If $\neg(P \vee Q) \in \mu$ then $\neg P \in \mu$ and $\neg Q \in \mu$

These are just the classical conditions. For modal logic we need the notion of a *model system*—a collection of model sets plus an alternativeness relation (actually, one for each knower a). The following additional rules govern the modal operators:

- (C.P*) If $\mathcal{P}_a P \in \mu$ and if μ belongs to a model system Ω , then there is in Ω at least one alternative μ^* to μ (with respect to a) such that $P \in \mu^*$
- (C.KK*) If $\mathcal{K}_a Q \in \mu$ and if μ^* is any alternative to μ (with respect to a) in model system Ω , then $\mathcal{K}_a Q \in \mu^*$
- (C.K) If $\mathcal{K}_a P \in \mu$ then $P \in \mu$
- (C. $\neg$ K) If $\neg \mathcal{K}_a P \in \mu$, then $\mathcal{P}_a \neg P \in \mu$
- (C. $\neg$ P) If $\neg \mathcal{P}_a P \in \mu$, then $\mathcal{K}_a \neg P \in \mu$

His proof procedure is by *reductio*: if the assumption that $\neg X$ belongs to a model set leads to an inconsistency, then X must belong to every model set, i.e., it must be true in all models, which is to say that it is valid. This has become the modern basis for tableau proofs.

Example 6.5.1 Show: $(\mathcal{K}_a P \wedge \mathcal{K}_a Q) \supset \mathcal{K}_a(P \wedge Q)$

1	$\mathcal{K}_a P \wedge \mathcal{K}_a Q \in \mu$	Assumption
2	$\neg \mathcal{K}_a(P \wedge Q) \in \mu$	Assumption
3	$\mathcal{P}_a \neg(P \wedge Q) \in \mu$	(C. $\neg$ K), 2
4	$\neg(P \wedge Q) \in \mu^*$	(C.P*), 3
5	$\mathcal{K}_a P \in \mu$	(C. $\wedge$), 1
6	$\mathcal{K}_a Q \in \mu$	(C. $\wedge$), 1
7	$\mathcal{K}_a P \in \mu^*$	(C.KK*), 5
8	$\mathcal{K}_a Q \in \mu^*$	(C.KK*), 6
9	$P \in \mu^*$	(C.K), 7
10	$Q \in \mu^*$	(C.K), 8
11	$\neg P \in \mu^*$	$\left \begin{array}{l} 12 \neg Q \in \mu^* \\ \times \end{array} \right.$
	$\times$	

In line 4, μ^* is an alternative model set to μ . Also lines 11 and 12 are the two alternatives deriving from line 4. One alternative is impossible, because of 9 and 11; the other is impossible, because of 10 and 12.

Epistemic logics continue to be of interest to philosophers. Nozick (1981) has raised some interesting problems with (6.2), $\mathcal{K}_a(P \supset Q) \supset (\mathcal{K}_a P \supset \mathcal{K}_a Q)$, in regard to skepticism. Let E be some experiential statement, e.g., “I see a human hand before me.” Let S be the crucial skeptical statement that I am unable to disprove, e.g., “I am dreaming” (or, alternatively, “I am a brain in a vat”). Now, it is quite clear that if E is true, i.e., if I *do* see a human hand before me (and not just something that appears to be a human hand), then S must be false, i.e., it cannot be the case that I am now dreaming. So, we have

$$E \supset \neg S, \quad (6.11)$$

and also, as just pointed out, we know this to be the case:

$$\mathcal{K}_a(E \supset \neg S). \quad (6.12)$$

Now, the skeptical position is just that I am unable to rule out the possibility that I am dreaming or that I am a brain in a vat. That is,

$$\neg \mathcal{K}_a \neg S. \quad (6.13)$$

An instance of (6.2) is

$$\mathcal{K}_a(E \supset \neg S) \supset (\mathcal{K}_a E \supset \mathcal{K}_a \neg S). \quad (6.14)$$

From this, contraposing the consequent, we have the following:

$$\mathcal{K}_a(E \supset \neg S) \supset (\neg \mathcal{K}_a \neg S \supset \neg \mathcal{K}_a E). \quad (6.15)$$

From (6.12) and (6.13), applying *Modus Ponens* twice, we derive

$$\neg \mathcal{K}_a E. \quad (6.16)$$

In effect, then, I have no experiential knowledge. Nozick (1981) suggests that we should reject (6.2). On the other hand, (6.16) could be correct. This interplay of philosophy and modal logic is a good example of how the sharpening provided by logic formalization enables us to grasp a philosophical puzzle.

6.5.3 The Knowability Paradox

The paradox was published as Fitch (1963). According to the account in the Stanford Encyclopedia, it stems from an anonymous referee report in 1945. In 2005 it was discovered that the referee was Alonzo Church. The report is reproduced in Church (2009) and discussed in Salerno (2009a). The Stanford article is a good general reference, Brogaard and Salerno (2019).

In brief, the paradox is this. If we assume that it is possible to know any truth, it follows that every truth is actually known. Of course this depends on what requirements we place on knowledge and of possibility. We will see what these are as we proceed.

As background, assume we have a propositional epistemic modal logic, with $\mathcal{K}$ as knowledge operator. Combine it with a modal logic with $\Box$ and $\Diamond$ as operators. We could, I suppose, think of these as metaphysical, though being metaphysical is not fundamental. Only minimal assumptions are needed. The epistemic part should be normal, and so has the rules and axioms of **K**, and it should include a factivity scheme: $\mathcal{K}X \supset X$. The modal operator should satisfy the Rule of Necessitation.

There is often special attention given to the scheme $X \wedge \neg \mathcal{K}X$. It is something one cannot know, because that would entail a contradiction by the following argument. Given our assumptions about normality of the epistemic logic, we know that $\mathcal{K}$ must distribute over conjunction, so

$$\begin{aligned} \mathcal{K}(X \wedge \neg \mathcal{K}X) &\supset (\mathcal{K}X \wedge \mathcal{K}\neg \mathcal{K}X) \\ &\supset (\mathcal{K}X \wedge \neg \mathcal{K}X) \end{aligned}$$

where the last step is by the factivity of knowledge. Since the consequent is contradictory, the following is provable.

$$\neg \mathcal{K}(X \wedge \neg \mathcal{K}X) \quad (6.17)$$

And then since we are assuming Necessitation for $\Box$, we have this:

$$\Box \neg \mathcal{K}(X \wedge \neg \mathcal{K} X). \quad (6.18)$$

It is important to note that in all of the above we were working, not with actual formulas, but with formula *schemes*. X could be any particular formula.

Now, following Fitch, suppose we adopt the following as a fundamental principle. Note that, again, this is a scheme and not a particular formula.

$$X \supset \Diamond \mathcal{K} X \quad (6.19)$$

Rather than the omniscience of asserting that each truth is known, $X \supset \mathcal{K} X$, formula (6.19) instead asserts that each truth is something that we can possibly know. This is more encouraging as a proposed axiom scheme. But then, one can reason as follows.

Suppose (as is likely) that there is some truth that we don't know. Let P_0 be such a truth. (This is an actual formula, and not a scheme.) Then we have the following.

$$P_0 \wedge \neg \mathcal{K} P_0 \quad (6.20)$$

Now (6.19) is a scheme, and we can take X to be (6.20), getting the following.

$$(P_0 \wedge \neg \mathcal{K} P_0) \supset \Diamond \mathcal{K} (P_0 \wedge \neg \mathcal{K} P_0) \quad (6.21)$$

Then from (6.20) and (6.21), using *Modus Ponens*, we have the following:

$$\Diamond \mathcal{K} (P_0 \wedge \neg \mathcal{K} P_0) \quad (6.22)$$

or equivalently

$$\neg \Box \neg \mathcal{K} (P_0 \wedge \neg \mathcal{K} P_0). \quad (6.23)$$

But (6.18) was a schema, so as a special case we have

$$\Box \neg \mathcal{K} (P_0 \wedge \neg \mathcal{K} P_0). \quad (6.24)$$

Obviously (6.23) and (6.24) contradict.

We got here by assuming there was some truth, P_0 , that we didn't know. So it must be that every truth is known! (At this point, see Exercise 6.6.3.)

6.6 Justification Logic

Hintikka style logics of knowledge work with assertions that an agent knows something, but not with the reasons for why this may be so. Starting in the 1990s with the work of Sergei Artemov, what is now called *Justification Logic* has evolved. This began with a justification counterpart of **S4** and important applications of it to the semantics of intuitionistic logic. The subject has since been extended to an infinite range of modal logics. In a justification logic instead of simple modal operators, one has a family of what are called *justification terms*, which one can informally understand as expressing reasons for something being known, or believed, or proved, etc. Instead of a simple $\Box X$ asserting that X is necessary, or known, or believed, or whatever, we have $t : X$ where t is a *justification term*, and this informally asserts that X is necessary, or known, or believed, *with t as its justification*.

The family of justification terms for a particular logic is built up using various operation symbols with the details depending on the particular justification logic. For example, something that is always present is an application operation, $\cdot$. Informally it is intended to represent an application of *Modus Ponens*. Thus, still speaking informally, if u justifies a formula X and t justifies $X \supset Y$, then $[t \cdot u]$ justifies Y . More formally, the following is an axiom schema of all justification logics.

$$t:(X \supset Y) \supset (u:X \supset [t \cdot u]:Y) \quad (6.25)$$

This is a justification counterpart of the modal validity

$$\Box(X \supset Y) \supset (\Box X \supset \Box Y) \quad (6.26)$$

and (6.25) is called a *realization* of (6.26). It provides a kind of analysis of (6.26): if we have $X \supset Y$ with t as its justification, and we have X with u as its justification, then these together with a *Modus Ponens* application will provide a justification for Y . As a more complex example, the modal validity

$$\Box(X \supset (Y \supset Z)) \supset (\Box Y \supset (\Box X \supset \Box Z)) \quad (6.27)$$

has the following realization (think about it).

$$a:(X \supset (Y \supset Z)) \supset (b:Y \supset (c:X \supset [(a \cdot c) \cdot b]:Z)) \quad (6.28)$$

One more operation symbol is generally added, $+$, intended to represent a kind of weakening. That is, if t is a justification for something, so are $t+u$ and $u+t$. Loosely, we can throw in extra stuff and still have a justification. Properly formulated (which we don't do here) this much gives us a justification counterpart of the modal logic

K. That is, one can prove that it is exactly the theorems of **K** that have provable realizations in the sense loosely described above.

Other modal logics have justification logics that correspond to them too. This applies to all the propositional modal logics we have introduced so far in this book, and in fact, the family of modal logics having justification versions is an infinite family. Justification logics provide a more nuanced view of what is going on in their corresponding modal logics. In a sense that can be made precise, justification terms serve to internalize proof machinery within the logic itself. This is especially important in a formal treatment of epistemic logics, where it is common to think of knowledge as being analyzable as *justified* true belief.

Beyond these very basic details, we do not present an analysis of justification logics here. A general discussion can be found in Artemov and Fitting (2011) and a full treatment is in Artemov and Fitting (2019). Both of these discuss applications of justification logics to philosophical issues and the latter especially covers a very broad range of modal/justification logic pairs. In addition, Kuznets and Studer (2019) concentrates on justification counterparts of modal logics from the modal cube.

Exercises

Exercise 6.6.1 Prove each of (6.7), (6.8) and (6.9) using Hintikka’s model sets.

Exercise 6.6.2 In the axiomatic system for classical logic from Sect. 2.7, all the axiom schemes are acceptable to an Intuitionist except for Excluded Middle, Scheme $(\neg\neg 2)$. Show that assuming these Intuitionistic axiom schemes plus Excluded Middle allows a proof of Double Negation Elimination, $\neg\neg X \supset X$. Hint: you will need $(\neg\neg 1)$ and $(\vee 3)$.

Exercise 6.6.3 The subsection about *The Knowability Paradox* was somewhat informal. For instance, the argument following (6.20) is really under the assumption of (6.20) as a hypothesis. The conclusion, “So, it must be that every truth is known!” is an assertion in English. Your problem is to turn the argument into a formal version. Explicitly state what your logical assumptions are. Wind up with the provability, under those assumptions, of the scheme $X \supset \mathcal{K} X$. Hint: show the equivalent $\neg(X \wedge \neg\mathcal{K} X)$.

References

- Artemov, S. N., & Fitting, M. (2011, revised 2015). Justification logic. Retrieved from <http://plato.stanford.edu/entries/logic-justification/>
- Artemov, S. N., & Fitting, M. (2019). *Justification logic: Reasoning with reasons*. Cambridge Tracts in Mathematics. Cambridge: Cambridge University Press.

- Blackburn, P., de Rijke, M., & Venema, Y. (2001). *Modal logic. Tracts in Theoretical Computer Science*. Cambridge: Cambridge University Press.
- Boolos, G. (1979). *The unprovability of consistency*. Cambridge: Cambridge University Press.
- Boolos, G. (1993). *The logic of provability*. Cambridge: Cambridge University Press. Paperback 1995.
- Brogaard, B., & Salerno, J. (2019). Fitch's paradox of knowability. Retrieved 2019, from <https://plato.stanford.edu/archives/fall2019/entries/fitch-paradox/>
- Chellas, B. (1980). *Modal logic*. Cambridge: Cambridge University Press.
- Church, A. (2009). Referee reports on Fitch's 'A Definition of Value'. In *Salerno (2009b)* (pp. 13–20).
- Fitch, F. (1963). A logical analysis of some value concepts. *Journal of Symbolic Logic*, 28(2), 135–142.
- Fitting, M. (1983). *Proof methods for modal and intuitionistic logics*. Dordrecht: D. Reidel.
- Gödel, K. (1933). Eine interpretation des intuitionistischen Aussagenkalküls. *Ergebnisse eines mathematischen Kolloquiums*, 4, 39–40. 'An Interpretation of the Intuitionistic Propositional Calculus' is translated in S. Feferman, ed., *Kurt Gödel, Collected Works, Volume One*, Oxford, pp. 300–303, 1986.
- Heyting, A. (1956). *Intuitionism, an introduction* (Second Edition 1965, Third Edition 1971). *Studies in Logic and the Foundations of Mathematics*. Amsterdam: North-Holland.
- Hintikka, J. (1962). *Knowledge and belief*. Ithaca: Cornell University Press.
- Kaplan, D. (1966). Review of Kripke. *The Journal of Symbolic Logic*, 31, 120–122.
- Kripke, S. (1959). A completeness theorem in modal logic. *Journal of Symbolic Logic*, 24, 1–14.
- Kripke, S. (1963a). Semantical analysis of modal logic I, normal propositional calculi. *Zeitschrift für mathematische Logik und Grundlagen der Mathematik*, 9, 67–96.
- Kripke, S. (1963b). Semantical considerations on modal logic. *Acta Philosophica Fennica*, 16, 83–94.
- Kripke, S. (1965). Semantical analysis of intuitionistic logic I. In J. N. Crossley & M. Dummett (Eds.), *Formal Systems and Recursive Functions, Proceedings of the Eight Logic Colloquium, Oxford 1963* (pp. 92–130). Amsterdam: North-Holland.
- Kuznets, R., & Studer, T. (2019). *Logics of proofs and justifications*. London: College Publications.
- Lemmon, E., & Scott, D. (1977). *An introduction to modal logic: The lemmon notes*. *American philosophical quarterly monograph series*. Oxford: B. Blackwell.
- Makinson, D. (1966). On some completeness theorems in modal logic. *Zeitschrift für mathematische Logik und Grundlagen der Mathematik*, 12, 379–384.
- Mints, G. (2012). The Gödel-Tarski translations of intuitionistic propositional formulas. In E. Eredem et al. (Ed.), *Correct reasoning. Lecture Notes in Computer Science* (7265, pp. 487–491). Berlin: Springer.
- Negri, S. (2009). Kripke completeness revisited. In G. Primiero & S. Rahman (Eds.), *Acts of knowledge—history, philosophy and logic* (pp. 247–282). London: College Publications.
- Nozick, R. (1981). *Philosophical explanations*. Cambridge: Harvard University Press.
- Salerno, J. (2009a). Knowability noir. In *Salerno (2009b)* (pp. 29–48).
- Salerno, J. (Ed.). (2009b). *New essays on the knowability paradox*. Oxford: Oxford University Press.
- Scott, D. S. (2022). Personal communication.

Chapter 7

Propositional Modal Tableaus



We discussed propositional classical tableaus in Chap. 3, and now this is extended to take modal operators into account. The extension retains the pleasant advantages that classical tableaus had. A tableau proof generally only uses subformulas and negations of subformulas of the formula being proved, and hence the search for a proof is easier than axiomatically. (There are exceptions to this: some logics require so-called *fixed point operators*, and these can violate the subformula property. This is not the case here.) Failed proof attempts, fully carried out, provide us with counter-examples. In fact, there are several kinds of modal tableaus that have been developed for this purpose. We will use what are called *prefixed tableaus*. These originated in Fitting (1972), and got the modular versions of the branch extension rules that are used here in Massacci (1998, 2000)—see also Goré (1998). (The survey article Fitting (1999) gives a historical overview that covers tableaus up to the date it was written.) We will discuss a few other kinds of tableaus at the end of the chapter.

7.1 Tableaus

One natural way of bringing modal concepts to bear in the design of tableau systems is to introduce explicit machinery to represent possible worlds. There are several ways of doing this. We make use of what have come to be known as prefixes. We first define them formally, then explain their intuition.

Definition 7.1.1 (Prefix) A *prefix* is a finite sequence of positive integers. A *prefixed formula* is an expression of the form σX , where σ is a prefix and X is a formula.

We will always write prefixes using periods to separate integers, 1.2.3.2.1 for instance. If σ is a prefix and n is a positive integer, $\sigma.n$ is σ followed by a period

followed by n . As you will see, tableau proofs begin with a formula prefixed with 1, and it follows from the tableau construction rules that all prefixes occurring in tableau proofs also begin with 1. But, nothing we will prove about the behavior of prefixes, primarily various versions of Hintikka's Lemma, requires that prefixes begin with 1. And in Sect. 16.5 more general prefixes make an appearance.

The intuitive idea is that a prefix σ names a possible world in some modal model, and σX tells us that X is true at the world that σ names. Our intention is that a prefix $\sigma.n$ should always name a world that is accessible from the one that σ names. For instance, if 1.2.1 is thought of as naming some possible world, then 1.2.1.1 and 1.2.1.2 and 1.2.1.3 and so on all name worlds accessible from that. Other facts about prefixes depend on which modal logic we are considering, but this is enough to get started.

An attempt to construct a tableau proof of a formula Z begins by creating a tree with $1 \neg Z$ at its root (and with no other nodes). Intuitively, the formula at the root says Z is false (that is, $\neg Z$ is true) at some world of some modal model, and we name that world with 1. Next, branches are "grown" according to certain *Branch Extension Rules*, to be given shortly. This yields a succession of tableaus for $1 \neg Z$. Finally, we need each branch to be *closed*, something we define after we give the rules for growing the branches.

Suppose we have $\sigma X \wedge Y$ at a node. Informally this says $X \wedge Y$ is true at the world that σ names, hence both X and Y are true there, and so we can extend the branch by adding σX and σY to the branch end. This is an example of a Branch Extension Rule. If we have a node with $\sigma X \vee Y$, informally $X \vee Y$ is true at the world that σ names. Then any branch going through this node can be split at the end with σX labeling the left child and σY labeling the right child. This is another Branch Extension Rule. Definition 7.1.2 contains the set of rules for the propositional connectives. They are the same as the classical ones in Definition 3.1.1, except for the prefixes.

Definition 7.1.2 (Prefixed Propositional Branch Extension Rules) In the following, σ is an arbitrary prefix.

CONJUNCTIVE RULES:

$$\frac{\sigma X \wedge Y}{\begin{array}{c} \sigma X \\ \sigma Y \end{array}} \quad \frac{\sigma \neg(X \vee Y)}{\sigma \neg X} \quad \frac{\sigma \neg(X \supset Y)}{\sigma \neg Y}$$

DISJUNCTIVE RULES:

$$\frac{\sigma X \vee Y}{\sigma X \mid \sigma Y} \quad \frac{\sigma \neg(X \wedge Y)}{\sigma \neg X \mid \sigma \neg Y} \quad \frac{\sigma X \supset Y}{\sigma \neg X \mid \sigma Y}$$

NEGATION RULE:

$$\frac{\sigma \neg \neg X}{\sigma X}$$

Now for the modal rules. Speaking informally, suppose we have $\sigma \Diamond X$, so at the world that σ names, $\Diamond X$ is true. Then there must be some world accessible from that one, at which X is true. By our naming convention, such a world should have a name of the form $\sigma.n$, but we shouldn't use a prefix of this form if it already has a meaning, since it might name the wrong world. The solution is to use $\sigma.n$, where

this prefix has never been used before. Then we are free to make it a name for a world at which X is true. This simple idea gives us the following rules.

Definition 7.1.3 (Prefixed Possibility Rules) If the prefix $\sigma.n$ is new to the branch,

$$\frac{\sigma \Diamond X}{\sigma.n X} \quad \frac{\sigma \neg \Box X}{\sigma.n \neg X}$$

Finally, consider $\sigma \Box X$. At the world that σ names, $\Box X$ is true, so at every world accessible from that one, X is true. It could happen that a prefix $\sigma.n$ doesn't name anything because we might not have used it, but if it does name, by our convention it names a world accessible from the one that σ names, so we should have $\sigma.n X$. This gives us the following rules.

Definition 7.1.4 (Prefixed Basic Necessity Rules) If the prefix $\sigma.n$ already occurs on the branch,

$$\frac{\sigma \Box X}{\sigma.n X} \quad \frac{\sigma \neg \Diamond X}{\sigma.n \neg X}$$

There is a strong similarity between these modal tableau rules and tableau rules for quantifiers, and this is no coincidence. Now we have the counterpart of Definition 3.1.2.

Definition 7.1.5 For modal propositional tableaus:

CLOSED BRANCH A modal tableau branch is *closed* if it contains both σX and $\sigma \neg X$ for some formula X , and if X is atomic, the branch is *atomically* closed. A branch that is not closed is *open*.

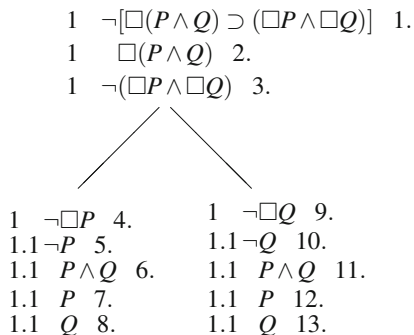
CLOSED TABLEAU A modal tableau is (*atomically*) *closed* if every branch is (atomically) closed.

TABLEAU PROOF A closed tableau for $1 \neg Z$ is a *tableau proof* of Z , and Z is a *theorem* if it has a tableau proof.

A closed branch obviously represents an impossible state of affairs—there is no modal model in which the prefixed formulas on it could all be realized. A closed tableau for $1 \neg Z$, in effect, says the consequences of assuming that Z could be false in some world of some model are contradictory.

We have seen semantic and axiomatic characterizations of several propositional modal logics, but (so far) only one tableau system. It is, in fact, a proof procedure for **K**. In the next section we give additional rules to produce tableau systems for other modal logics. But first an example.

Example 7.1.6 According to Example 5.3.6, $\Box(P \wedge Q) \supset (\Box P \wedge \Box Q)$ is **K**-valid. Here is a tableau proof of the formula.



In this tableau item 1 is, of course, how the proof begins. Items 2 and 3 are from 1 by a Conjunctive Rule; 4 and 9 are from 3 by a Disjunctive Rule; 5 is from 4 by a Possibility Rule (note, the prefix 1.1 is new on the branch at this point); 6 is from 2 by a Basic Necessity Rule (note, the prefix 1.1 already occurs on the branch at this point); 7 and 8 are from 6 by a Conjunctive Rule. The left branch is closed because of 5 and 7. The right branch has a similar explanation. Please note that, though 1.1 occurs on both branches, in each case it is new when introduced *on the branch*. Branches are independent cases and what happens on one has no effect on another.

Exercises

Exercise 7.1.1 In the tableau system given above, for the modal logic **K**, prove the following.

1. $(\Box P \wedge \Box Q) \supset \Box(P \wedge Q)$
2. $\Box(P \supset Q) \supset (\Box P \supset \Box Q)$
3. $(\Box P \wedge \Diamond Q) \supset \Diamond(P \wedge Q)$
4. $(\Box \Diamond P \wedge \Box \Box Q) \supset \Box \Diamond(P \wedge Q)$

Exercise 7.1.2 Give a tableau proof of $(\Box P \vee \Box Q) \supset \Box(P \vee Q)$. On the other hand the converse, $\Box(P \vee Q) \supset (\Box P \vee \Box Q)$, is not provable. Explain why a tableau proof is impossible. Discuss the intuitive reasons why one would not want this formula to be provable.

Exercise 7.1.3 Show, using the **K** tableau system, that if $\Box X$ has a tableau proof, so does X .

7.2 More Tableau Systems

In the previous section we gave a single tableau system, which we will show in Sect. 7.6 proves exactly the valid formulas of **K**. Now we add various rules to that system to produce proof procedures for a wider range of modal logics. Prefixed tableau systems are considerably less general than axiom systems, but they do exist for the entirety of the modal cube. When we come to the first-order material that this book is primarily about, our tableau systems will be a central tool. But, for philosophical applications, only a few of the modal logics we have seen turn out to be actually of interest. These are among the ones in the modal cube with what we have referred to as ‘historical’ names.

In the interests of presenting a relatively uncomplicated tableau treatment now, we pare down the collection of logics we are discussing by removing three of them: **KB4**, **KD5**, and **K5**. The reasons for removing these three are as follows. The tableau rules we use, often called *single step* rules, come from Massacci (1998, 2000), and Goré (1998). Especially in Massacci (2000), rules and detailed analyses are given for the entire of the modal cube. Three of the rules, there called 4^π , 4^D , and Cxt , are somewhat less natural (personal opinion) than the others and we prefer not to make use of them here. The rule Cxt is needed for **K5** and **KD5**, so these are dropped. The rule 4^π is used for **KB4**, **K45**, **KD45**, and **S5**, but all of **K45**, **KD45**, and **S5** have alternative formulations that do not need the rule, so of these only **KB4** is dropped. This reduces the modal cube, Fig. 5.1, by removing three logics and produces a simpler structure which we call the *Lesser Modal Cube*. It is displayed in Fig. 7.1. Only tableau rules for the logics it contains are considered here. In fact eventually our discussion will be reduced further to those logics with historical names, and these are listed first in Fig. 7.2, followed by rules appropriate for the rest of the Lesser Modal Cube.

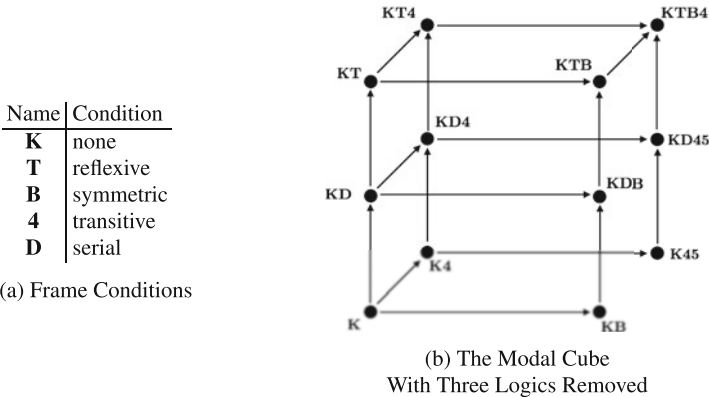


Fig. 7.1 The lesser modal cube

Fig. 7.2 Special tableau rules for the lesser modal cube

Logic	Tableau Rules Added to K
K	none
T = KT	T
S4 = KT4	$T, 4$
B = KTB	T, B
S5 = KTB4	$T, 4, 4r$
D = KD	D
K4	4
KB	B
K45	$4, 4r$
KD4	$D, 4$
KDB	D, B
KD45	$D, 4, 4r$

It is important to note that, for the approach we are taking in the later chapters of this book, any tableau system that provides some kind of explicit representation of possible worlds should work as well. In particular, this is the case for the ones discussed in Sect. 7.7. Our prefixed tableau version only handles modal logics that can be characterized by models having a tree structure (though we make no use of this fact). The ones in Sect. 7.7 are, in that sense, more general than ours, and could be used instead. But none of this is fundamentally significant for understanding our applications to quantified modal logics.

Here are the various tableau rules that will be added to the rules for **K** that were given in Sect. 7.1.

Definition 7.2.1 (Prefixed Special Necessity Rules) For prefixes σ and $\sigma.n$ already occurring on a tableau branch:

$$\begin{array}{lcl}
 T & \frac{\sigma \Box X}{\sigma X} & \frac{\sigma \neg \Diamond X}{\sigma \neg X} \\
 D & \frac{\sigma \Box X}{\sigma \Diamond X} & \frac{\sigma \neg \Diamond X}{\sigma \neg \Box X} \\
 B & \frac{\sigma.n \Box X}{\sigma X} & \frac{\sigma.n \neg \Diamond X}{\sigma \neg X}
 \end{array}
 \qquad
 \begin{array}{lcl}
 4 & \frac{\sigma \Box X}{\sigma.n \Box X} & \frac{\sigma \neg \Diamond X}{\sigma.n \neg \Diamond X} \\
 4r & \frac{\sigma.n \Box X}{\sigma \Box X} & \frac{\sigma.n \neg \Diamond X}{\sigma \neg \Diamond X}
 \end{array}$$

Now for each of the logics of the Lesser Modal Cube, we get a tableau system by adding to the rules for **K** various combinations of the Special Necessity Rules. This is shown in Fig. 7.2, beginning with the most common modal logics.

We show a few very simple examples of proofs using these rules, then a somewhat more complicated one.

Example 7.2.2 Here is a proof, using the **KB** rules, of $\Diamond\Box X \supset X$. As usual, numbers to the right of each item are for reference only, and are not an official part of the tableau proof.

$$\begin{array}{ll} 1 & \neg(\Diamond\Box X \supset X) \quad 1. \\ 1 & \Diamond\Box X \quad 2. \\ 1 & \neg X \quad 3. \\ 1.1 & \Box X \quad 4. \\ 1 & X \quad 5. \end{array}$$

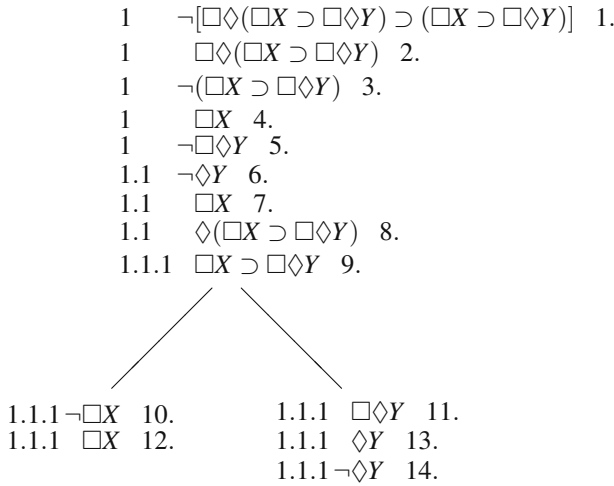
Formula 1 is the usual starting point. Formulas 2 and 3 are from 1 by a Conjunctive Rule; formula 4 is from 2 by a Possibility Rule; then 5 is from 4 by Special Necessity Rule *B*. Closure is by 3 and 5.

Example 7.2.3 The following is a proof, using the **T** rules, of $[\Box(X \vee Y) \wedge \neg X] \supset Y$.

$$\begin{array}{ll} 1 & \neg\{[\Box(X \vee Y) \wedge \neg X] \supset Y\} \quad 1. \\ 1 & \Box(X \vee Y) \wedge \neg X \quad 2. \\ 1 & \neg Y \quad 3. \\ 1 & \Box(X \vee Y) \quad 4. \\ 1 & \neg X \quad 5. \\ 1 & X \vee Y \quad 6. \\ & \swarrow \quad \searrow \\ 1X & 7. \quad \quad 1Y \quad 8. \end{array}$$

Formulas 2 and 3 are from 1 by a Conjunctive Rule; 4 and 5 are from 2 the same way; formula 6 is from 4 by Special Necessity Rule *T*; 7 and 8 are from 6 by a Disjunctive Rule. Closure is by 5 and 7 on the left branch, and 3 and 8 on the right.

Example 7.2.4 Here is a proof of greater complexity, using the **S4** rules, of the formula $\Box\Diamond(\Box X \supset \Box\Diamond Y) \supset (\Box X \supset \Box\Diamond Y)$.

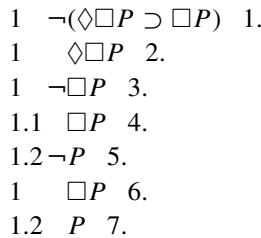


Formula 1 is the usual starting point. Formulas 2 and 3 are from 1 by a Conjunctive Rule; 4 and 5 are from 3 also by a Conjunctive Rule; 6 is from 5 by a Possibility Rule; 7 is from 4 by Special Necessity Rule 4; 8 is from 2 by a Basic Necessity Rule; 9 is from 8 by a Possibility Rule; 10 and 11 are from 9 by a Disjunctive Rule; 12 is from 7 by Special Necessity Rule 4; 13 is from 11 by Special Necessity Rule *T*; and 14 is from 6 by Special Necessity Rule 4. The left branch is closed because of 10 and 12; the right branch because of 13 and 14.

We point out that this tableau, while closed, is not *atomically* closed. You should see that it would be easy to extend it to one that is.

Finally we consider the special case of **S5**, which actually has an alternative and much simpler system, as we will see below. Here is an example that uses the tableau rules from Fig. 7.2.

Example 7.2.5 Using the **S5** rules, here is a proof of $\Diamond\Box P \supset \Box P$.



In this, 2 and 3 are from 1 by a Conjunctive Rule; 4 is from 2 by a Possibility Rule, as is 5 from 3; 6 is from 4 by Special Necessity Rule 4*r*; 7 is from 6 by a Basic Necessity Rule. Notice that we could have stopped without line 7, because 3 and 6 already close the branch. With 7 added, the closure is at the *atomic* level. It is

a feature of the tableau systems we give, that closure can always be brought to the atomic level. (This is not the case with tableau systems for all modal logics, but such logics are not among those we look at here.)

While there is certainly nothing wrong with the **S5** tableau system used in Example 7.2.5, there is a greatly simpler version. Take as prefixes just *positive integers*, and not sequences of them. And replace all the rules by the following.

Definition 7.2.6 (Simplified S5 Tableau Rules) Non-modal rules are as before, except that prefixes are all of length one. The modal rules are the following.

S5 POSSIBILITY RULE If the integer k is new to the branch,

$$\frac{n \Diamond X}{k X} \quad \frac{n \neg \Box X}{k \neg X}$$

S5 NECESSITY RULE If the integer k already occurs on the branch,

$$\frac{n \Box X}{k X} \quad \frac{n \neg \Diamond X}{k \neg X}$$

We conclude the section with an example using this simplified **S5** system.

Example 7.2.7 We give a proof of $\Box P \vee \Box \neg \Box P$.

$$\begin{array}{ll} 1 \neg(\Box P \vee \Box \neg \Box P) & 1. \\ 1 \neg \Box P & 2. \\ 1 \neg \Box \neg \Box P & 3. \\ 2 \neg P & 4. \\ 3 \neg \neg \Box P & 5. \\ 3 \Box P & 6. \\ 2 P & 7. \end{array}$$

Formulas 2 and 3 are from 1 by a Conjunctive Rule; 4 is from 2 by a Possibility Rule; 5 is from 3 similarly; 6 is from 5 by the Negation Rule; 7 is from 6 by a Necessity Rule.

Exercises

Exercise 7.2.1 Prove $\Diamond(P \supset \Box P)$ in the **T** system.

Exercise 7.2.2 Prove $(\Box X \wedge \Box Y) \supset \Box(\Box X \wedge \Box Y)$ in the **K4** system.

Exercise 7.2.3 Prove $(\Box \Diamond X \wedge \Box \Diamond Y) \supset \Box \Diamond(\Box \Diamond X \wedge \Box \Diamond Y)$ in the **S4** system. Can it be proved using the **K4** rules?

Exercise 7.2.4 Prove $\Box P \vee \Box(\Box P \supset Q)$ in each of the **S5** systems.

Exercise 7.2.5 Give an **S5** proof of $\Box(\Diamond P \supset P) \supset \Box(\Diamond \neg P \supset \neg P)$.

Exercise 7.2.6 Prove $\Diamond(P \wedge \Box Q) \equiv (\Diamond P \wedge \Box Q)$ in each of the **S5** systems. That is, separately prove an implication each way.

Exercise 7.2.7 One can embed classical logic into both **S5** and **S4**. For the first, put $\Box$ in front of every subformula. For the second, put $\Box\Diamond$. The following use these embeddings of the tautology $P \vee \neg P$.

1. Give an **S5** tableau proof of $\Box(\Box P \vee \Box\neg\Box P)$.
2. Give an **S4** tableau proof of $\Box\Diamond(\Box\Diamond P \vee \Box\Diamond\neg\Box\Diamond P)$.

7.3 Logical Consequence and Tableaus

We have already discussed logical consequence semantically in a modal setting, $S \models_{\mathbf{L}} U \rightarrow X$, in Sect. 5.6, and we discussed it axiomatically in Sect. 6.3. We have seen that it is a more complex thing than its classical counterpart because it involves both global assumptions, S , and local assumptions, U . Recall, the semantic idea was that X should be true at each world of a model at which the members of U are all true, provided the members of S are true at every world of that model. Fortunately, tableau rules for this notion of consequence are both simple and intuitive.

Definition 7.3.1 (Tableau Assumption Rules) Let $\mathbf{L}$ be one of the modal logics of the Lesser Modal Cube, so we have tableau rules for it. Also let S and U be sets of formulas. A tableau uses S as *global* assumptions and U as *local* assumptions if the following two additional tableau rules are admitted.

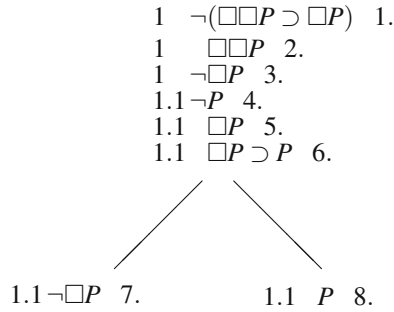
LOCAL ASSUMPTION RULE If Y is any member of U then $1 Y$ can be added to the end of any open branch.

GLOBAL ASSUMPTION RULE If Y is any member of S then σY can be added to the end of any open branch on which σ appears as a prefix.

Thus local assumptions can be used with prefixes of 1 , and 1 is the prefix associated with the formula we are trying to prove—local assumptions can only be used locally, so to speak. Likewise any prefix that has appeared can be used with a global assumption, corresponding to the idea that global assumptions are available at every world.

If assumptions, local or global, are involved in a tableau construction for $1 \neg X$, we refer to the tableau as a *derivation* of X rather than a proof of X .

Example 7.3.2 In Example 5.6.3 we showed semantically that $\{\Box P \supset P\} \models_{\mathbf{K}} \emptyset \rightarrow \Box\Box P \supset \Box P$ holds. Now we give a **K** derivation of $\Box\Box P \supset \Box P$ using $\{\Box P \supset P\}$ as the set of global assumptions.



Items 2 and 3 are from 1 by a Conjunctive Rule; 4 is from 3 by a Possibility Rule; 5 is from 2 by a Necessity Rule; 6 uses the Global Assumption Rule; 7 and 8 are from 6 by a Disjunctive Rule.

Exercises

Exercise 7.3.1 Give a **K** derivation of $\Box\Box P \supset P$ using $\Box P \supset P$ as a global assumption. Compare it to a **K** derivation of P using $\Box P \supset P$ as a global assumption and $\Box\Box P$ as a local assumption.

7.4 Modal Tableau Soundness

We gave a possible world semantics for logics in the Modal Cube in Chap. 5, and axiomatic proof systems for them in Chap. 6. We showed the axiom systems were sound and complete with respect to the semantics. Now we prove similar results for tableau systems for logics in the Lesser Modal Cube—soundness in this section and completeness in the next. Once we do this we will also have established that our two kinds of proof procedures prove the same formulas for modal logics that are in the Lesser Modal Cube, namely the ones valid in the corresponding possible world semantics.

We follow the same ideas that worked for classical logic in Sect. 3.3. We introduce an appropriate notion of what it means for a modal tableau to be satisfiable; we then show that closed tableaux are not satisfiable, and that satisfiability is preserved by each of the tableau rules. One thing that is new here is the presence of prefixes, and the definition of satisfiability must take them into account. The other new thing is that the language has been extended to include modal operators; we have tableau rules for them, and satisfiability preservation must be shown to extend to these new rules as well.

Prefixes are meant to be something like possible worlds, with accessibility represented via syntactic structure. The following definition formalizes the informal phrase “something like”.

Definition 7.4.1 (Prefix Function) Let S be a set of prefixed modal formulas, and let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ be a modal model. A *prefix function* for S in $\mathcal{M}$ is a mapping, call it θ , assigning to each prefix that occurs in S some possible world in $\mathcal{G}$ so that, if σ and $\sigma.n$ both occur in S then $\theta(\sigma.n)$ is a world accessible from $\theta(\sigma)$, that is, $\theta(\sigma)\mathcal{R}\theta(\sigma.n)$.

Definition 7.4.2 (Satisfiable) Let $\mathbf{L}$ be one of the logics from the Lesser Modal Cube. A set S of prefixed modal formulas is *satisfiable* in an $\mathbf{L}$ model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ if there is a prefix mapping for S in $\mathcal{M}$ so that whenever σX is in S , then X is true at the world $\theta(\sigma)$, that is,

$$\sigma X \in S \implies \theta(\sigma) \Vdash X.$$

A tableau branch is $\mathbf{L}$ satisfiable if the set of prefixed formulas on it is satisfiable in some $\mathbf{L}$ model. A tableau is $\mathbf{L}$ satisfiable if some branch of it is $\mathbf{L}$ satisfiable.

The two fundamental results about classical propositional satisfiability are in Propositions 3.3.2 and 3.3.3. We need modal counterparts of these. Here is the first.

Proposition 7.4.3 *A closed $\mathbf{L}$ tableau is not $\mathbf{L}$ satisfiable.*

The proof is an easy modification of that for Propositions 3.3.2 and we leave it as Exercise 7.6.1. The second result needs a little more work to prove.

Proposition 7.4.4 *For $\mathbf{L}$ being one of the modal logics in the Lesser Modal Cube, if an $\mathbf{L}$ tableau branch extension rule is applied to an $\mathbf{L}$ satisfiable tableau, the result is another $\mathbf{L}$ satisfiable tableau.*

Proof Suppose $\mathcal{T}$ is a satisfiable $\mathbf{L}$ tableau, and a branch extension rule is applied to it; say on branch $\mathcal{B}$. If the tableau has a branch other than $\mathcal{B}$ that is satisfiable, it remains satisfiable after the rule application. This is a trivial case. So now suppose a tableau rule is applied to $\mathcal{B}$ itself. If the rule is one of the propositional connective ones, the argument is almost the same as in the proof of Proposition 3.3.3, except that prefixes now occur. We leave this to you as Exercise 7.6.2. So, for the rest of the proof suppose one of the modal rules is applied to a prefixed formula on satisfiable branch $\mathcal{B}$. There are three major cases to consider. The first two are common to all choices for $\mathbf{L}$. The third depends on which particular modal logic $\mathbf{L}$ happens to be.

For the rest of the proof, assume branch $\mathcal{B}$ of tableau $\mathcal{T}$ is $\mathbf{L}$ satisfiable. Say the set of prefixed formulas on it is $\mathbf{L}$ satisfiable in the model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$, using the mapping θ from prefixes occurring on $\mathcal{B}$ to worlds in $\mathcal{G}$. In each case the tableau rules come in pairs and we present the argument for one of the two. The other is analogous in each case.

Prefixed Possibility Rules, Definition 7.1.3 Suppose $\sigma \Diamond X$ occurs on $\mathcal{B}$ and we add $\sigma.n X$ to the end of $\mathcal{B}$, where the prefix $\sigma.n$ does not occur on the branch

$\mathcal{B}$, and so $\theta(\sigma.n)$ is not currently defined. Call the extended branch $\mathcal{B}'$. By our assumptions about the satisfiability of $\mathcal{B}$, we know that $\theta(\sigma) \Vdash \Diamond X$ in the model $\mathcal{M}$. But then for some possible world Δ in $\mathcal{G}$, $\theta(\sigma) \mathcal{R} \Delta$ and $\Delta \Vdash X$. Define a new mapping, θ' as follows. For all the prefixes occurring on $\mathcal{B}$, set θ' to be the same as θ . Since $\sigma.n$ did not occur on $\mathcal{B}$, we are free to define θ' on it as we please, and we set $\theta'(\sigma.n)$ to be Δ .

Since θ and θ' agree on the prefixes of $\mathcal{B}$, all the prefixed formulas of $\mathcal{B}$ have the same behavior in $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ no matter whether we use θ or θ' . Further, $\theta'(\sigma) \mathcal{R} \theta'(\sigma.n)$ since $\theta'(\sigma) = \theta(\sigma)$, $\theta'(\sigma.n) = \Delta$, and $\theta(\sigma) \mathcal{R} \Delta$. And finally, $\theta'(\sigma.n) \Vdash X$. It follows that all prefixed formulas on the branch $\mathcal{B}'$ are satisfiable in $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$, using the mapping θ' .

Prefix Basic Necessity Rules, Definition 7.1.4 Suppose $\sigma \Box X$ occurs on $\mathcal{B}$ and also prefix $\sigma.n$ appears on $\mathcal{B}$. And suppose we add $\sigma.n X$ to the end of $\mathcal{B}$, using a Basic Necessity Rule. Call the resulting branch $\mathcal{B}'$. We show $\mathcal{B}'$ is satisfiable.

Since $\mathcal{B}$ is satisfiable we have a mapping θ meeting the conditions of Definition 7.4.3. The prefixes appearing on $\mathcal{B}'$ are the same as those on $\mathcal{B}$, so θ does not need changing. We show it meets the appropriate conditions for the larger branch $\mathcal{B}'$ as well. Very simply, since θ satisfies $\mathcal{B}$ in the model $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$, it must be that $\theta(\sigma) \mathcal{R} \theta(\sigma.n)$, and also $\theta(\sigma) \Vdash \Box X$. But then $\theta(\sigma.n) \Vdash X$, so θ also satisfies the extended branch $\mathcal{B}'$ in $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$.

Prefix Special Necessity Rules, Definition 7.2.1 We have that **L** is one of the logics in the Lesser Modal Cube, from Fig. 7.1. For this case, assume we add a prefixed formula to the branch $\mathcal{B}$ of tableau $\mathcal{T}$ using one of the Special Rules from Fig. 7.2. We will show that the extended branch is still satisfied in the model $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$. Rather than considering all eleven logics, it is enough to look at the five cases corresponding to the specific type of rule from Definition 7.2.1 that was applied. As usual, we will consider only one of the pair of rules making up each case. And we will actually only consider two of the cases, leaving the rest as exercises. The cases we discuss are rules 4 and 4r.

Case for Rule *T*: Rule *T* preserves satisfiability in a reflexive model. Exercise.

Case for Rule *D*: Rule *D* preserves satisfiability in a serial model. Exercise.

Case for Rule *B*: Rule *B* preserves satisfiability in a symmetric model. Exercise.

Case for Rule 4: We show Rule 4 preserves satisfiability if $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ is transitive. Suppose $\sigma \Box X$ is on $\mathcal{B}$ and we add $\sigma.n \Box X$ to $\mathcal{B}$ to get $\mathcal{B}'$, where $\sigma.n$ already occurs in $\mathcal{B}$. We show that in $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$, $\theta(\sigma.n) \Vdash \Box X$. Suppose Δ is an arbitrary world in $\mathcal{G}$ such that $\theta(\sigma.n) \mathcal{R} \Delta$. We have that $\theta(\sigma) \mathcal{R} \theta(\sigma.n)$, so by transitivity, $\theta(\sigma) \mathcal{R} \Delta$. Since $\theta(\sigma) \Vdash \Box X$ then $\Delta \Vdash X$. And since Δ was arbitrary, $\theta(\sigma.n) \Vdash \Box X$.

Case for Rule 4r: This time we show that Rule 4r preserves satisfiability provided $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ is symmetric and transitive. Suppose $\sigma.n \Box X$ occurs on branch $\mathcal{B}$, σ occurs as a prefix somewhere on $\mathcal{B}$, and branch $\mathcal{B}'$ is the result of adding

$\sigma \Box X$ to the end of $\mathcal{B}$. Also assume $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ has a symmetric and transitive frame, and that branch $\mathcal{B}$ is satisfiable in this model using the mapping θ . Then in particular, $\theta(\sigma.n) \Vdash \Box X$. To show that $\mathcal{B}'$ is satisfiable in the model, it is enough to show that $\theta(\sigma) \Vdash \Box X$. And for this it is enough to show that for any $\Delta \in \mathcal{G}$ such that $\theta(\sigma)\mathcal{R}\Delta$ we have that $\Delta \Vdash X$.

Well, $\theta(\sigma)\mathcal{R}\theta(\sigma.n)$ by Definition 7.4.2. Since the model frame is symmetric, $\theta(\sigma.n)\mathcal{R}\theta(\sigma)$. Then by transitivity, $\theta(\sigma.n)\mathcal{R}\Delta$. Since $\theta(\sigma.n) \Vdash \Box X$, then $\Delta \Vdash X$, which is what we wanted.

We proved soundness for classical propositional tableaus, as Theorem 3.3.4. With small changes in wording, the same proof applies here.

Theorem 7.4.5 (Modal Tableau Soundness) *For $\mathbf{L}$ being one of the modal logics in the Lesser Modal Cube, if X has a tableau proof using the $\mathbf{L}$ rules, X is $\mathbf{L}$ -valid.*

Proof Assume X is not a validity of the modal logic $\mathbf{L}$. Then there is an $\mathbf{L}$ model $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$, and a possible world $\Gamma \in \mathcal{G}$ so that $\Gamma \not\Vdash X$. We show this implies there can be no $\mathbf{L}$ proof for X .

To try to prove X , we start a tableau with $1 \neg X$. Let θ be the (rather trivial) mapping such that $\theta(1) = \Gamma$. Using θ , our initial tableau is $\mathbf{L}$ satisfiable in the model $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$. Beginning with a satisfiable tableau, every tableau that we subsequently construct will also be $\mathbf{L}$ satisfiable, by Proposition 7.4.4, so we can never construct a closed tableau by Proposition 7.4.3. So no tableau proof of X using the $\mathbf{L}$ rules is possible.

Exercises

Exercise 7.4.1 Complete the proof of Proposition 7.4.4 by showing the three missing cases in the proof, for Rules T , D , and B .

7.5 Modal Hintikka Sets

Throughout this section, $\mathbf{L}$ is one of the modal logics in the Lesser Modal Cube. In Sect. 3.4 we gave two proofs of completeness for classical propositional tableaus. Both have analogs for $\mathbf{L}$. The common part of both proofs was the use of Hintikka Sets, Definition 3.4.1, and Hintikka's Lemma, Proposition 3.4.2. There are versions of these for prefixed tableaus, which we now introduce. In the present context a Hintikka set for a logic is, roughly speaking, a set of prefixed formulas that is closed under the tableau rules for the logic. Here are the details. Note that parts 1 and 2 in the Hintikka set definition below are exactly as in the classical setting of Definition 3.4.1, except for the presence of prefixes which play no role for these

parts. It is only in parts 3 and 4 that prefix structure becomes significant. The relationship with Definition 7.2.1 and Fig. 7.2 should be obvious.

Definition 7.5.1 (Hintikka Set) A set H of prefixed propositional modal formulas is an **L Hintikka set** if it meets the following four conditions, where σ is an arbitrary prefix.

1. H meets the consistency condition; it does not contain both σA and $\sigma \neg A$ for any atomic formula A .
2. H meets the following propositional closure conditions:

$$\begin{aligned}
 \sigma X \wedge Y \in H &\implies \sigma X \in H \text{ and } \sigma Y \in H \\
 \sigma \neg(X \wedge Y) \in H &\implies \sigma \neg X \in H \text{ or } \sigma \neg Y \in H \\
 \sigma X \vee Y \in H &\implies \sigma X \in H \text{ or } \sigma Y \in H \\
 \sigma \neg(X \vee Y) \in H &\implies \sigma \neg X \in H \text{ and } \sigma \neg Y \in H \\
 \sigma X \supset Y \in H &\implies \sigma \neg X \in H \text{ or } \sigma Y \in H \\
 \sigma \neg(X \supset Y) \in H &\implies \sigma X \in H \text{ and } \sigma \neg Y \in H \\
 \sigma \neg\neg X \in H &\implies \sigma X \in H
 \end{aligned}$$

3. H meets the following modal closure conditions:

$$\begin{aligned}
 \sigma \Box X \in H &\implies \sigma.n X \in H \text{ for every } \sigma.n \text{ appearing in } H \\
 \sigma \neg\Diamond X \in H &\implies \sigma.n \neg X \in H \text{ for every } \sigma.n \text{ appearing in } H \\
 \sigma \Diamond X \in H &\implies \sigma.n X \in H \text{ for some } \sigma.n \\
 \sigma \neg\Box X \in H &\implies \sigma.n \neg X \in H \text{ for some } \sigma.n
 \end{aligned}$$

4. H meets certain special closure conditions, depending on the choice of **L**. We first state conditions, then say which go with which logic.

Condition	Definition
T	$\sigma \Box X \in H \implies \sigma X \in H$ and $\sigma \neg\Diamond X \in H \implies \sigma \neg X \in H$
D	$\sigma \Box X \in H \implies \sigma \Diamond X \in H$ and $\sigma \neg\Diamond X \in H \implies \sigma \neg\Box X \in H$
B	$\sigma.n \Box X \in H \implies \sigma X \in H$ and $\sigma.n \neg\Diamond X \in H \implies \sigma \neg X \in H$
4	$\sigma \Box X \in H \implies \sigma.n \Box X \in H$ and $\sigma \neg\Diamond X \in H \implies \sigma.n \neg\Diamond X \in H$ provided $\sigma.n$ appears in H
$4r$	$\sigma.n \Box X \in H \implies \sigma \Box X \in H$ and $\sigma.n \neg\Diamond X \in H \implies \sigma \neg\Diamond X \in H$

Closure conditions assigned to each logic.

Logic L	Conditions on H
K	none
T	T
B	T, B
S4	$T, 4$
S5	$T, 4, 4r$
D	D
K4	4
KB	B
K45	$4, 4r$
KD4	$D, 4$
KDB	D, B
KD45	$D, 4, 4r$

Hintikka sets can be small simple things. For instance, the empty set is a trivial Hintikka set for any of the modal logics. So is any set whose only members are all prefixed formulas of the form σP , where P is atomic and σ is any prefix. This is an example of an infinite Hintikka set. Here is a **K** Hintikka set that is a little more interesting.

$$\{1 (\Box \Diamond P \wedge \Diamond Q), 1 \Box \Diamond P, 1 \Diamond Q, 1.1 Q, 1.1 \Diamond P, 1.1.1 P\}$$

Infinite Hintikka sets can be more complex than the trivial atomic example above, and sometimes they cannot be avoided when rules 4 and 4r are involved. Suppose, for instance, that the logic is **K4**, and we want a Hintikka set containing $1 \neg(\Box X \vee \Diamond \Box X)$. Even though a finite Hintikka set containing this prefixed formula does exist, the construction rules we gave for **K4** will produce an infinite one. We will see more about this particular Hintikka set when we come to Example 7.6.4. Nonetheless, even with all the complications we still have modal versions of Hintikka's Lemma.

Proposition 7.5.2 (Hintikka's Lemma) *Every L Hintikka set H is L satisfiable in an L model.*

Proof Let H be an L Hintikka set. We construct an L model in which H is satisfied. We will first do this assuming L is **K**, then we will discuss how to modify the construction for two other logics in the Lesser Modal Cube. We leave the rest to you. A common feature across all the logics is that the model we construct will have as its set of possible worlds the collection of prefixes that occur in H . For each logic, once a model $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ has been constructed, we will show that H is satisfied in it by showing the following by Complete Induction on the complexity of the formula Z , where our measure of complexity is the degree of the formula, Definition 5.1.2.

$$\begin{aligned} &\text{if } \sigma Z \in H \text{ then } \sigma \Vdash Z \\ &\text{if } \sigma \neg Z \in H \text{ then } \sigma \not\Vdash Z \end{aligned} \tag{★★}$$

This can be seen as a generalization of ($\star$), which came up in our proof of a version of Hintikka's Lemma for classical propositional logic, Proposition 3.4.2.

Now we begin. The argument differs in some details from logic to logic. We start with **K**.

Assume H is a **K Hintikka Set** We build a model in which H is satisfied.

Let $\mathcal{G}$ be the collection of prefixes that occur in the set H . If σ and $\sigma.n$ are both in $\mathcal{G}$, set $\sigma \mathcal{R} \sigma.n$. Finally, if P is a propositional letter, and σP occurs in H , take P to be true at σ , that is, $\sigma \models P$. Otherwise take P to be false at σ . This completely determines a model $\langle \mathcal{G}, \mathcal{R}, \models \rangle$.

Now we will show ($\star\star$) by Complete Induction. Assume Z is a formula and ($\star\star$) is known for formulas of lower degree. There are several cases, depending on the form of Z .

Suppose first that Z is of degree 0. Then it is a propositional variable, say P . If $\sigma P \in H$, $\sigma \models P$ by definition. And if $\sigma \neg P \in H$, $\sigma P \notin H$ by Condition 1 of Definition 7.5.1, so $\sigma \not\models P$, again by definition.

From here on Z is of degree bigger than 0. If Z matches one of the propositional cases in Condition 2 of Definition 7.5.1, the argument is the same as that in the proof of Proposition 3.4.2, except for the presence of prefixes (which play no active role here). We do not repeat the earlier argument.

The modal cases are the new thing now. There are four of them, two for $\Box$ and two for $\Diamond$. The $\Diamond$ cases are similar to the $\Box$ ones, and are omitted.

Suppose that Z is $\Box X$, where the induction hypothesis ($\star\star$) is known for formulas of lower degree, in particular for X . If $\sigma \Box X \in H$ then also $\sigma.n X \in H$ for every prefix $\sigma.n$ that appears in H , by Condition 3 of Definition 7.5.1. By the induction hypothesis, $\sigma.n \models X$ for every prefix $\sigma.n$ that occurs in H . But this is equivalent to saying that $\sigma' \models X$ for every $\sigma' \in \mathcal{G}$ such that $\sigma \mathcal{R} \sigma'$. It follows that $\sigma \models \Box X$. Similarly, if $\sigma \neg \Box X \in H$ for some σ then $\sigma.n \neg X$ also occurs, for some prefix $\sigma.n$. Thus there is some $\sigma' \in \mathcal{G}$ (namely $\sigma.n$) such that $\sigma \mathcal{R} \sigma'$ and $\sigma' \not\models X$ (using the induction hypothesis). So $\sigma \not\models \Box X$.

This ends the argument if **L** is **K**. We have shown that each formula in H is satisfied in the model we constructed and, since **K** places no conditions on the accessibility relation of an **K** model, the model constructed is one for **K**.

Assume H is an **S4 Hintikka Set** Construct an **S4** model in which H is satisfied. Exactly as in the **K** case, let $\mathcal{G}$ be the set of prefixes in H , and for each propositional letter P take P to be true at σ , that is, $\sigma \models P$, just if $\sigma P \in H$. There is an important difference from the previous case when it comes to defining the accessibility relation $\mathcal{R}$. For **K** no special conditions were needed, but now we need reflexivity and transitivity, which must be introduced somehow.

We begin by defining a relation $\mathcal{R}_0$ as follows. For $\Gamma, \Delta \in \mathcal{G}$, set $\Gamma \mathcal{R}_0 \Delta$ if $\Gamma = \sigma$ and $\Delta = \sigma.n$ for some prefix σ and number n , or if Γ and Δ are the same prefix. The first condition embodies the basic idea concerning the prefix structure, and was all that was needed for **K**. The second adds reflexivity to the mix. And now we make use of $\mathcal{R}_0$ to define the accessibility relation we really want, $\mathcal{R}$, by

bringing what is called transitive closure in. For $\Gamma, \Delta \in \mathcal{G}$ let $\Gamma \mathcal{R} \Delta$ if there is a sequence $\sigma_1, \sigma_2, \dots, \sigma_k$ of members of $\mathcal{G}$ such that:

1. $\sigma_1 = \Gamma$,
2. $\sigma_k = \Delta$,
3. and for each $i = 1, \dots, k - 1$ we have $\sigma_i \mathcal{R}_0 \sigma_{i+1}$.

It is easy to show that the relation $\mathcal{R}$ is transitive. Suppose $\Gamma \mathcal{R} \Delta$ and $\Delta \mathcal{R} \Omega$. Then there is a sequence of members of $\mathcal{G}$ going from Γ to Δ with each item in it related to the next using $\mathcal{R}_0$. There is a similar sequence going from Δ to Ω . The end of the first sequence is the start of the second. ‘Glue’ the two sequences together at Δ . The result is another sequence in which each item is related to the next one using $\mathcal{R}_0$, and the sequence takes us from Γ through Δ to Ω , so $\Gamma \mathcal{R} \Omega$. Also it is almost trivial that $\mathcal{R}$ is reflexive. We thus have an **S4** model $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$.

The central matter now is to show that, given our somewhat elaborate definition of $\mathcal{R}$, we have the key fact that for $\Gamma, \Delta \in \mathcal{G}$, if $\Gamma \Box X \in H$ and $\Gamma \mathcal{R} \Delta$ then $\Delta X \in H$. We work our way up to this through a couple of preliminary results.

First, suppose $\Gamma, \Delta \in \mathcal{G}$ and $\Gamma \mathcal{R} \Delta$. Let $\sigma_1, \sigma_2, \dots, \sigma_k$ be a sequence of members of $\mathcal{G}$ meeting conditions 1–3 as set out above, taking us from $\Gamma = \sigma_1$ to $\Delta = \sigma_k$, with the relation $\mathcal{R}_0$ holding between each member of the sequence and the next. If $\sigma_i \Box X \in H$ for any $i = 1, \dots, k - 1$, then $\sigma_{i+1} \Box X \in H$. The argument for this has two cases that need to be checked, coming from the two cases in the definition of $\mathcal{R}_0$. First, it could be that $\sigma_i = \sigma_{i+1}$; this case is trivial. Second, it could be that $\sigma_{i+1} = \sigma_i.n$ for some n . In this case we can use Condition 4 from part 4 of Definition 7.5.1, which applies to **S4**.

Next, suppose we know that $\sigma_1 \Box X \in H$. It follows that $\sigma_k X \in H$. This is because we can repeatedly apply the result we just showed to conclude that $\sigma_k \Box X \in H$, and then $\sigma_k X \in H$ follows by Condition *T* from part 4 of Definition 7.5.1, which applies to **S4**. We have now showed that if $\Gamma \mathcal{R} \Delta$ and $\Gamma \Box X \in H$, then $\Delta X \in H$.

We are finally ready to prove an **S4** version of $(\star\star)$. This will establish that our **S4** Hintikka set H is satisfied in the **S4** model $\langle \mathcal{G}, \mathcal{R}, \Vdash \rangle$ that we have constructed. In fact almost the entire of the earlier argument for **K** applies—it is only the modal case that needs some work. Suppose Z is $\Box X$, and $(\star\star)$ is known to hold for X . Suppose $\Gamma \Box X \in H$, for some $\Gamma \in \mathcal{G}$. We have just shown that for any $\Delta \in \mathcal{G}$, if $\Gamma \mathcal{R} \Delta$ then $\Delta X \in H$. Then by the induction hypothesis, $\Delta \Vdash X$. Since Δ was arbitrary, it follows that $\Gamma \Vdash \Box X$. The argument for the case where $\sigma \neg \Box X \in H$ is exactly as in the **K** case. The cases involving $\Diamond$ are similar and are omitted. This ends the argument for the **S4** case.

Assume H is a KB4 Hintikka Set We construct a model in which H is satisfied. As in the **S4** case, the main problem is to make sure we construct a model that has the features required by the logic, in this case it must be symmetric and transitive. And to do this, we proceed in a way that is quite similar to the **S4** argument. The set of possible worlds of the model, $\mathcal{G}$, is again the collection of prefixes occurring in H . An atomic formula P is true at possible world σ , that is, $\sigma \Vdash P$, just when $\sigma P \in H$.

A relation $\mathcal{R}_0$ is defined on $\mathcal{G}$ similarly to before, except that now we do not want reflexivity but we do want symmetry. For $\Gamma, \Delta \in \mathcal{G}$, $\Gamma \mathcal{R}_0 \Delta$ if $\Gamma = \sigma$ and $\Delta = \sigma.n$ for some n , or else if $\Gamma = \sigma.n$ and $\Delta = \sigma$ for some n . Extending this to transitive relation $\mathcal{R}$ is done as before—we repeat the definition for convenience. For $\Gamma, \Delta \in \mathcal{G}$ let $\Gamma \mathcal{R} \Delta$ if there is a sequence $\sigma_1, \sigma_2, \dots, \sigma_k$ of members of $\mathcal{G}$, where k is at least 2, such that:

1. $\sigma_1 = \Gamma$,
2. $\sigma_k = \Delta$,
3. and for each $i = 1, \dots, k - 1$ we have $\sigma_i \mathcal{R}_0 \sigma_{i+1}$.

The same argument as in the **S4** case shows that $\mathcal{R}$ is transitive. We leave it to you to check that $\mathcal{R}$ is also symmetric. What remains to be shown is that if $\Gamma \Box X \in H$ and $\Gamma \mathcal{R} \Delta$, then $\Delta X \in H$, and then the rest of the argument is exactly as it was for **S4**. Checking that this is so concludes the proof.

Let $\sigma_1, \sigma_2, \dots, \sigma_k$ be a sequence of members of $\mathcal{G}$ meeting conditions 1–3, with the relation $\mathcal{R}_0$ holding between any two consecutive members of the sequence. We first show that for each $i = 1, 2, \dots, k - 1$, if $\sigma_i \Box X \in H$, then both $\sigma_{i+1} \Box X$ and $\sigma_{i+1} X$ are in H . So, assume $\sigma_i \Box X \in H$. There are two cases, coming from the two cases in the definition of $\mathcal{R}_0$. First, we might have that $\sigma_{i+1} = \sigma_i.n$ for some n . If so, Part 3 and Condition 4 from Part 4 of Definition 7.5.1, both of which apply to **KB4**, give us what we need. Second, we might have that $\sigma_i = \sigma_{i+1}.n$ for some n . For this situation, Conditions *B* and 4 from Part 4 of Definition 7.5.1 take care of things, and both of these apply to **KB4**.

Other Logics We have given details for three of the logics from the Lesser Modal Cube. That leaves nine more logics to be checked. These are similar and we omit the proofs. We do give a few cases as exercises, though.

7.6 Propositional Modal Tableau Completeness

In Sect. 3.4 we proved completeness for a classical tableau system. Similar methods apply to prefixed tableaus for modal logics. Classically we began by introducing Hintikka sets, and we have now done a similar thing for modal logics. And as we did classically, we have now shown that Hintikka sets are satisfiable for the modal logics that concern us here. The final step is to extract Hintikka sets from tableaus. Classically we did that twice, once constructively and once non-constructively. The same two approaches apply here, and we present both. And as we did classically, we begin with the constructive approach.

7.6.1 Modal Tableau Completeness, Constructively

The essence of this approach to completeness involves finding a *fair* algorithm for constructing tableaus. What it means formally to be a fair algorithm is stated in Definition 3.4.3, but loosely the idea is, a tableau construction algorithm is fair if it eventually does each particular thing that the rules allow to be done.

Describing a fair tableau construction algorithm can be tricky, depending on the choice of logic. Here is an informal presentation of one for **K**, the simplest case. The algorithm proceeds in stages. As it executes, some tableau branches will get marked *finished*, and some prefixed formula occurrences will get marked *used*. If at some stage all branches are marked *finished*, the algorithm terminates. It can be shown for **K** that the algorithm below must terminate. For some other logics, termination may not happen, the algorithm proceeds indefinitely, and an infinite tableau is generated. We can ignore this complication here.

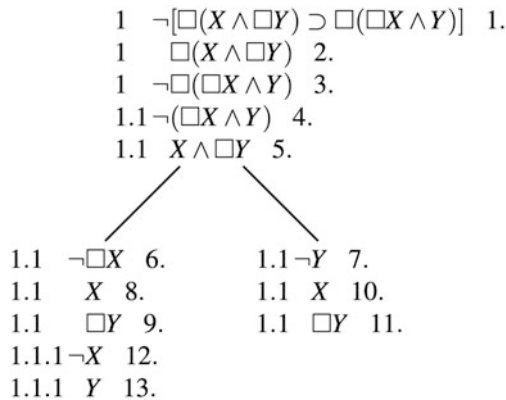
Suppose we start constructing a **K** tableau for $1 \neg X$. At the start we have a one-branch tableau, with only a single prefixed formula occurrence on it. The branch is not marked as *finished*, and the prefixed formula occurrence on it is not marked *used*. This ends stage 1.

Suppose stage n of the tableau construction algorithm has been completed. If every branch is marked *finished*, the tableau construction terminates. Otherwise, for stage $n + 1$ we process each branch that is not marked *finished*, beginning with the leftmost branch not marked *finished*. Then we process the first branch to its right not marked *finished*, then the next to the right, and so on until we have worked with the rightmost branch. This ends stage $n + 1$.

A branch is processed in stage $n + 1$ as follows. If the branch is atomically closed, mark it as *finished*. If the branch is not atomically closed, and it contains a prefixed formula that is not marked *used*, where that formula is not a necessity formula, or the negation of a possibility formula, or an atomic formula or the negation of an atomic formula, then pick the uppermost such formula node, and apply the appropriate rule to it on each branch that goes through that formula node, and then mark that prefixed formula occurrence *used*. After this, whether or not some prefixed formula had a rule applied to it as just described, for each necessity formula $\sigma \Box X$ on the branch, and for each prefix of the form $\sigma.n$ that occurs on the branch, add $\sigma.n X$ to the branch if it is not already there. Similarly for each negated possibility formula $\sigma \neg \Diamond X$ on the branch. If no formulas were added in either the first or the second part of this process, mark the branch as *finished*.

We worked with uppermost unused formulas above. But it can be shown that, no matter how the choice of a formula to work with is made, the process must terminate. Verifying this depends on the fact that every **K** tableau rule, when applied, results in prefixed formulas being added that are simpler than the one to which the rule was applied. We do not prove termination here. We do, however, give an illustrative example of a tableau that has been constructed following this procedure.

Example 7.6.1 The following is an attempted proof of $\Box(X \wedge \Box Y) \supset \Box(\Box X \wedge Y)$ using the **K** rules, following the fair construction procedure for **K** that was sketched above.



At the start, formula 1 is the only unused one, so we apply a Conjunctive Rule to it, getting 2 and 3, after which 1 has been used. Since 2 is a necessity type formula, we cannot choose it; instead we apply a Possibility Rule to 3, getting 4 and leaving 3 now used. After this we can apply a Necessity Rule to 2, getting 5. The uppermost unused formula now is 4 and applying a rule to it yields 6 and 7. Now the uppermost unused formula is 5, and this adds 8 and 9 to the left branch and also 10 and 11 to the right, after which it is used. On the left branch, the uppermost unused formula is 6, and applying a rule to this yields 12 after which 9 yields 13. No new rule applications apply to the right branch. At this point the construction algorithm terminates.

We will continue with this example below. But first, at this point we have everything we need to prove completeness for the **K** tableau system.

Theorem 7.6.2 (Propositional K Tableau Completeness) *If X is **K**-valid, X has a tableau proof using the **K** rules. In fact, X will have a tableau proof that is atomically closed, and can be found by following a fair tableau construction algorithm.*

Proof Suppose modal formula X does not have a **K** tableau proof. Then a fair tableau construction starting with $1 \neg X$ cannot close. It is not hard to see that for any tableau proof attempt that is fairly constructed, the set of prefixed formulas on any tableau branch that remains open must be a **K** Hintikka set. Then by Hintikka's Lemma, Proposition 7.5.2, the set of prefixed formulas on an open branch is satisfiable in some **K** model. Since an open branch, like every branch of the tableau, must contain $1 \neg X$, there will be a **K** model with a possible world (corresponding to prefix 1) at which $\neg X$ is true, and hence X is false. Thus X is not **K** valid.

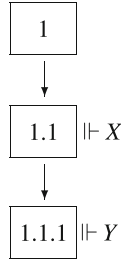
Example 7.6.3 We continue Example 7.6.1, using the model construction procedure outlined in the **K** part of the proof of Proposition 7.5.2. Both tableau branches

in the example are open—say we work with the left one. We construct a model $\langle \mathcal{G}, \mathcal{R}, \models \rangle$ as follows.

First, we form the set $\mathcal{H}$ of prefixed formulas on the left branch. This is a Hintikka set, and is as follows.

$$\begin{aligned} \mathcal{H} = \{ & 1 \neg[\Box(X \wedge \Box Y) \supset \Box(\Box X \wedge Y)], 1 \Box(X \wedge \Box Y), 1 \neg\Box(\Box X \wedge Y), \\ & 1.1 \neg(\Box X \wedge Y), 1.1 X \wedge \Box Y, 1.1 \neg\Box X, 1.1 X, 1.1 \Box Y, 1.1.1 \neg X, 1.1.1 Y \} \end{aligned}$$

Let $\mathcal{G} = \{1, 1.1, 1.1.1\}$, the set of prefixes occurring in $\mathcal{H}$. This is the set of possible worlds of the model we are constructing. Let $1\mathcal{R}1.1$ and $1.1\mathcal{R}1.1.1$. Finally, set $1.1 \models X$, $1.1.1 \models Y$, and in no other cases are propositional letters true at worlds. We now have a model $\langle \mathcal{G}, \mathcal{R}, \models \rangle$, displayed in schematic form below.



The prefixed formula $1.1 \neg\Box X$ is in $\mathcal{H}$. And in fact, $1.1 \not\models \Box X$ in the model since $1.1\mathcal{R}1.1.1$ and $1.1.1 \not\models X$. Similarly $1.1 \Box Y$ is in $\mathcal{H}$, and $1.1 \models \Box Y$ since the only possible world of the model that is accessible from 1.1 is 1.1.1, and we have $1.1.1 \models Y$. In this way we can verify that the set of prefixed formulas in $\mathcal{H}$ is satisfiable in the model we constructed. In particular, $\Box(X \wedge \Box Y) \supset \Box(\Box X \wedge Y)$ is not true at world 1, and hence is not valid.

We have now shown completeness of the tableau system for **K**. The other modal tableau systems for logics from the Lesser Modal Cube have similar completeness proofs, but with added complications, all stemming from the need to describe a fair tableau construction procedure for tableaus following the rules of that logic. As an example, consider the rules for **K4**; if we apply a fair tableau construction procedure we discover that it may never terminate!

Example 7.6.4 Suppose we try to prove $\Box X \vee \Diamond\Box X$ in **K4**, and so start a tableau with its negation. We now proceed as follows.

- 1 $\neg(\Box X \vee \Diamond\Box X)$ 1.
- 1 $\neg\Box X$ 2.
- 1 $\neg\Diamond\Box X$ 3.
- 1.1 $\neg X$ 4.
- 1.1 $\neg\Box X$ 5.
- 1.1 $\neg\Diamond\Box X$ 6.

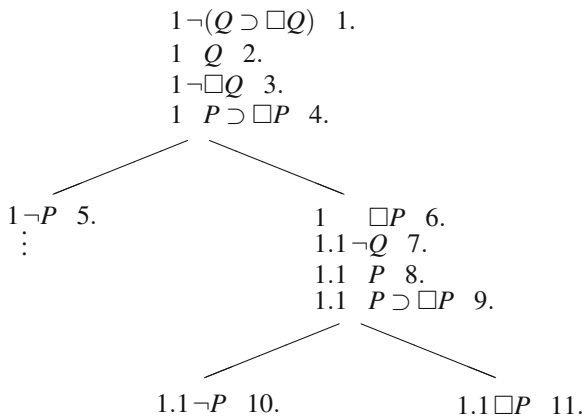
Formulas 2 and 3, of course, are from 1 by a Conjunctive Rule. Formula 4 is from 2 by a Possibility Rule, and then 5 is from 3 by a Necessity Rule. But now, in **K4**, Rule 4 has us add formula 6 using formula 3, and a desire to apply all applicable rules means we must do so. Now, formulas 5 and 6 duplicate formulas 2 and 3, but with lengthened prefixes. We can work with 5 and 6 the same way we did with 2 and 3, getting fresh occurrences of the same formulas, with 1.1.1 prefixes. And so on. In short, we find ourselves in an infinite loop, generating an infinite branch with longer and longer prefixes, but repeating the same formulas.

The possibility of infinite loops is no problem for completeness proofs. It simply means we construct infinite, not finite, models. But we must be careful with our tableau constructions to ensure that every potentially applicable rule is actually applied. This takes some care, and we omit a proper discussion here. See Fitting (1983) or Goré (1998) for details.

7.6.2 Logical Consequence

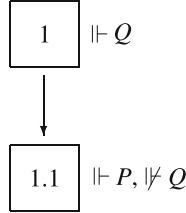
We have shown, in Sect. 7.3, how to modify tableau systems to use global and local assumptions. Extending soundness and completeness arguments to take these into account is quite straightforward. Rather than giving the full details we give one illustrative example, and leave the rest to you.

We attempt a **K** derivation of $Q \supset \Box Q$ from $P \supset \Box P$ as a global assumption. As we will see, this attempt fails, and from the failed attempt we will be able to construct a model showing it is not the case that $\{P \supset \Box P\} \models_{\mathbf{K}} \emptyset \rightarrow Q \supset \Box Q$, or using abbreviated notation, it is not the case that $\{P \supset \Box P\} \models_{\mathbf{K}} Q \supset \Box Q$.



Items 2 and 3 are from 1 by a Conjunctive Rule; 4 is a Global Assumption; 5 and 6 are from 4 by a Disjunctive Rule; 7 is from 3 by a Possibility Rule; 8 is from 6 by a Necessity Rule; 9 is a Global Assumption; 10 and 11 are from 9 by a

Disjunctive Rule. The branch ending with item 10 is closed. The branch through 5 is not finished yet, but the branch ending with 11 is not closed, and every applicable rule has been applied on it. We use it to construct our counter-model. Doing as the branch instructs, we get the following.



We leave it to you to check that every formula on the branch ending with 1.1 is satisfied in this model and the model is, in fact, the counter-model we wanted. Note in particular that the global assumption $P \supset \Box P$ is true at every possible world in this model. It is true at world 1 because P is not true there. And it is true at world 1.1 because $\Box P$ is true there.

7.6.3 Modal Completeness, Non-constructively

In Sect. 3.4 we gave two different completeness proofs for classical propositional tableaux, and both made use of Hintikka sets. The difference was that in one version a Hintikka set was constructed during the course of a failed attempt at finding a tableau proof, while in the other it was shown to exist as a maximal consistent set in a tableau sense. The first version was constructive, the second was not. Both kinds of proofs exist for modal propositional tableaux too. We have just seen the first kind, using a proof attempt involving steps specified by a fair construction algorithm. As you may have noticed, fair construction algorithms can be hard to formulate. In this section we present the second modal approach, which avoids this difficulty but does not establish decidability of a logic.

There is one obvious difference between tableaux for classical logic and prefixed modal tableaux: the prefixes. And the key feature of this difference is embodied in Definition 7.1.3: conclude $\sigma.n X$ from $\sigma \Diamond X$ *provided* $\sigma.n$ is new to the branch. This has a strong resemblance to the way existential quantifiers are treated in natural deduction systems, or in informal mathematical reasoning too for that matter. Henkin (1949) introduced the now-standard way of proving completeness for classical first-order logic by making use of sets of formulas that are not only maximally consistent, but are also *E-complete*, where E-completeness amounts to saying we have enough names in the language to instantiate the existential quantifiers we come across. This is the idea behind the machinery here as well, and so we mention Henkin explicitly in our terminology. The following is our modal analog of Definition 3.4.6 for classical tableaux.

Definition 7.6.5 (Tableau Consistency) In this definition $\mathbf{L}$ is one of the logics in the Lesser Modal Cube, and so $\mathbf{L}$ tableaux are tableaux following the $\mathbf{L}$ rules from Fig. 7.2.

Let S be a set of prefixed propositional modal formulas. If S is finite, it is *$\mathbf{L}$ tableau consistent* if no atomically closed $\mathbf{L}$ tableau beginning with the members of S exists. If S is infinite, it is *$\mathbf{L}$ tableau consistent* if every finite subset is. S is *maximally $\mathbf{L}$ tableau consistent* if it is $\mathbf{L}$ tableau consistent, but no proper extension is $\mathbf{L}$ tableau consistent. S is *$\mathbf{L}$ tableau inconsistent* if it is not $\mathbf{L}$ tableau consistent.

Before we go on, there is a hidden point that needs some clarification. In our tableau Possibility Rules, Definition 7.1.3, if $\sigma \Diamond X$ is on a tableau branch, $\sigma.n X$ can be added to the branch end *provided $\sigma.n$ is new to the branch*. What *new* means is clear in tableau proofs, where all prefixes that appear are built up from 1 by adding one number at a time during the course of the tableau construction. But we now are considering tableaux that start with an arbitrary finite set of prefixed formulas, and these prefixes have not been built up during the course of a proof. So to be clear, if 1.2.1.3 occurs on a branch we will say that all of 1, 1.2, 1.2.1, and 1.2.1.3 are present and do not count as new.

As we noted above, a set of quantified formulas is often called *E-complete* if it contains a witness for each existentially quantified formula present, and similarly for negated universal formulas. By analogy we introduce the terminology *P-complete* for a set of prefixed formulas that contains a witness for each possibility formula present, and similarly for negated necessities.

Definition 7.6.6 (P-Complete) A prefixed formula is *of type P* if it is one of the forms $\sigma \Diamond X$ or $\sigma \neg \Box X$. A set S of prefixed formulas is *P-complete* provided,

$$\begin{aligned} \sigma \Diamond X \in S &\implies \sigma.k X \in S \text{ for some integer } k, \\ \sigma \neg \Box X \in S &\implies \sigma.k \neg X \in S \text{ for some integer } k. \end{aligned}$$

Theorem 7.6.7 (Modal Lindenbaum-Henkin) Any set S of prefixed formulas that is $\mathbf{L}$ tableau consistent and contains only a finite number of prefixes can be extended to a maximally $\mathbf{L}$ tableau consistent set that is *P-complete*.

Proof The proof builds on that of Theorem 3.4.7, which in turn was a tableau version of Theorem 2.4.7. The elaboration consists in modifying things to force *P* completeness to hold. We give enough details here so that the earlier proofs should not need to be consulted, but it wouldn't hurt to do so.

Let S be a set of prefixed formulas that is $\mathbf{L}$ tableau consistent, and that contains a finite number of prefixes. The set of all prefixed formulas is countable—assume they have been enumerated as: $Z_1, Z_2, \dots$ (Please note that these are *prefixed* formulas, and not just plain formulas). Now recursively define the following sequence of sets.

$$S_1 = S$$

$$S_{n+1} = \begin{cases} S_n \cup \{Z_n\} & \text{if } S_n \cup \{Z_n\} \text{ is } \mathbf{L} \text{ tableau consistent} \\ & \text{and } Z_n \text{ is not of type } P, \\ S_n \cup \{Z_n, \sigma.k X\} & \text{if } S_n \cup \{Z_n\} \text{ is } \mathbf{L} \text{ tableau consistent,} \\ & Z_n \text{ is } \sigma \Diamond X, \text{ and } k \text{ is the smallest number} \\ & \text{for which } \sigma.k \text{ does not appear in } S_n, \\ S_n \cup \{Z_n, \sigma.k \neg X\} & \text{if } S_n \cup \{Z_n\} \text{ is } \mathbf{L} \text{ tableau consistent,} \\ & Z_n \text{ is } \sigma \neg \Box X, \text{ and } k \text{ is the smallest number} \\ & \text{for which } \sigma.k \text{ does not appear in } S_n, \\ S_n & \text{otherwise} \end{cases}$$

Since $S = S_1$ contains finitely many prefixes, it follows easily that every S_n does the same. Then in the second and third cases, where prefixed formulas of type P are involved, there will always be some new prefix $\sigma.k$ available, so these cases can be carried out when appropriate.

In the first and the last of the four cases defining S_{n+1} , it is trivial that S_{n+1} is $\mathbf{L}$ tableau consistent provided S_n is. In the two cases where type P prefixed formulas come up, a small argument is needed. We discuss one of the two such cases, where $Z = \sigma \Diamond X$. The other is similar.

We want to show that if $S_n \cup \{\sigma \Diamond X\}$ is $\mathbf{L}$ tableau consistent; then $S_{n+1} = S_n \cup \{\sigma \Diamond X, \sigma.k X\}$ is also $\mathbf{L}$ tableau consistent, provided $\sigma.k$ is not in S_n . We do this by showing a converse version. We assume that $\sigma.k$ is not in S_n but $S_{n+1} = S_n \cup \{\sigma \Diamond X, \sigma.k X\}$ is $\mathbf{L}$ tableau inconsistent, and we show it follows that $S_n \cup \{\sigma \Diamond X\}$ is $\mathbf{L}$ tableau inconsistent.

By assumption there is a closed $\mathbf{L}$ tableau starting with a finite subset of S_{n+1} , that is, with a finite subset of $S_n \cup \{\sigma \Diamond X, \sigma.k X\}$. Then there is a finite set $F \subseteq S_n$ such that there is a closed $\mathbf{L}$ tableau starting with the members of $F \cup \{\sigma \Diamond X, \sigma.k X\}$; call this tableau $\mathcal{T}$. Now start a new tableau beginning with the members of $F \cup \{\sigma \Diamond X\}$. We apply a Prefixed Possibility Rule, Definition 7.1.3, adding $\sigma.k X$, which we can do because $\sigma.k$ is not in S_n , and hence not in any prefixed formula in $F \cup \{\sigma \Diamond X\}$. We can now continue the tableau to closure by just copying the steps from tableau $\mathcal{T}$. This tells us that $S_n \cup \{\sigma \Diamond X\}$ is $\mathbf{L}$ tableau inconsistent.

We now have that if S_n is $\mathbf{L}$ tableau consistent so is S_{n+1} . Since S_1 is $\mathbf{L}$ tableau consistent, it follows that every member of the sequence is $\mathbf{L}$ tableau consistent. Let $S_\infty = S_1 \cup S_2 \cup S_3 \cup \dots$. If S_∞ were not $\mathbf{L}$ tableau consistent, it would have a finite subset F that was not. A finite $F \subseteq S_\infty$ must be a finite subset of S_n for some big enough value of n . But then S_n would be $\mathbf{L}$ tableau inconsistent, and it is not. Thus S_∞ is $\mathbf{L}$ tableau consistent, and it is not hard to check that it is maximally so. By its construction, S_∞ is obviously P -complete, ending the proof.

Maximally $\mathbf{L}$ tableau consistent, P complete sets of prefixed formulas have exactly the properties we want. This is easily summarized in the following.

Theorem 7.6.8 *If M is a maximally $\mathbf{L}$ consistent, P -complete set of prefixed formulas, then M is an $\mathbf{L}$ Hintikka set.*

Proof Let M be a maximally $\mathbf{L}$ consistent, P -complete set. The conditions for an $\mathbf{L}$ Hintikka set are given in Definition 7.5.1, and it must be shown that M meets these conditions. The classical propositional cases, 1 and 2, are essentially the same as in the proof of Proposition 3.4.2, but with prefixes added. We leave these to you.

From part 3 of the Hintikka set definition, we only look at the first item. We show that if $\sigma \Box X \in M$ and the prefix $\sigma.n$ occurs in M then $\sigma.n X \in M$. We actually show the contrapositive. Assume that $\sigma.n$ occurs in M but $\sigma.n X$ does not belong to M . Since M is maximal, adding $\sigma.n X$ to M must be $\mathbf{L}$ tableau inconsistent. Then there must be a finite subset $F \subseteq M$ such that an $\mathbf{L}$ tableau starting with the prefixed formulas in $F \cup \{\sigma.n X\}$ closes. Since $\sigma.n$ occurs in M , we can make sure we have a prefixed formula with it as a prefix among the formulas of F . But now there must be a closed $\mathbf{L}$ tableau starting with the members of $F \cup \{\sigma \Box X\}$. This is because we can make the first step the addition of $\sigma.n X$ to the initial tableau branch, using a Prefixed Basic Necessity Rule. and then continue with the tableau steps for the closed $\mathbf{L}$ tableau for $F \cup \{\sigma.n X\}$. This tells us that $\sigma \Box X$ cannot be in M since the set would be $\mathbf{L}$ tableau inconsistent.

Part 4 of the Hintikka set definition gives a number of conditions depending on the particular choice for $\mathbf{L}$. These are handled exactly as in the previous case, and details are omitted.

We now have a second proof of completeness for prefixed tableaux. Suppose X is not tableau provable using the rules for $\mathbf{L}$. Then there is no closed $\mathbf{L}$ tableau for $1 \neg X$, so the set $\{1 \neg X\}$ is consistent. Extend it to a maximally $\mathbf{L}$ tableau consistent, P -complete set M . This set is an $\mathbf{L}$ Hintikka set and hence by Proposition 7.5.2 it is $\mathbf{L}$ satisfiable. Then there is an $\mathbf{L}$ model and a world in it, corresponding to the prefix 1, at which $\neg X$ is true, so X is not $\mathbf{L}$ valid.

Exercises

Exercise 7.6.1 Give a full proof of Proposition 7.4.3 using the proof of Proposition 3.3.2 as a guide.

Exercise 7.6.2 Give proofs for the binary cases of Proposition 7.4.4, modeled on the proof of Proposition 3.3.3 as a guide.

Exercise 7.6.3 Give detailed soundness proofs for the tableau system for (some of) **D**, **T**, **K4**, **B**, **S4** and for both tableau systems for **S5**.

Exercise 7.6.4 Construct a model corresponding to the right branch of the tableau in Example 7.6.1, and show in detail that $\Box(X \wedge \Box Y) \supset \Box(\Box X \wedge Y)$ fails at a world of it.

Exercise 7.6.5 Continue with the pattern of construction begun in Example 7.6.4, and use the infinite tableau that results to produce a **K4** model in which $\Box X \vee \Diamond \Box X$ is not valid.

Exercise 7.6.6 The formula $\Box(\Diamond P \supset P) \supset \Box(\Diamond \neg P \supset \neg P)$ is not provable in **S4**, though it is in **S5** (see Exercise 7.2.5). Extract an **S4** model showing it is not valid from a failed attempt to prove it using a systematic tableau construction.

Exercise 7.6.7 Define a suitable notion of soundness for tableaux that use the Local and Global Assumption Rules, and prove the **K** tableau system with these rules is, in fact, sound.

Exercise 7.6.8 Attempt a **K** derivation of $\Box P \supset \Box \Box P$ from $\Box P \supset P$ as a global assumption. From the failed attempt, construct a model showing it is not the case that $\{\Box P \supset P\} \models_{\mathbf{K}} \Box P \supset \Box \Box P$. As it happens, the tableau construction does not terminate, but fortunately after a certain point there is enough regularity to see a pattern.

7.7 Other Kinds of Tableaus

When we come to the central topics of this book we will make much use of tableaux. Axiomatics also applies, but it is much harder to see what is going on when used. Tableaus are closer to our intuitions. In particular we will lean heavily on the presence of prefixes in our tableau systems. These provide a way of dealing explicitly with possible worlds using our proof machinery.

Not all modal tableau systems involve extra machinery like prefixes, but such proof systems cover a smaller range of modal logics. In the other direction, there are tableau mechanisms that provide the means to represent explicit possible worlds, but not by using prefixes. Prefixes encode the accessibility relation structurally into the prefixes themselves, but are outside the actual modal language. Other methodologies exist that do something equivalent, but *internally*. That is, the machinery of formulas is enlarged in ways that allow direct talk about possible worlds. In this section we sketch three tableau systems that you can find in the literature. Any of these could have been used for the main work that is coming up in later parts of this book. The ideas we will present are not tied to a particular formalism. We hope our brief sketches will help you better understand what our main points are, and separate these points from the exact details of the machinery we use to make the points.

The three systems we discuss form a kind of hierarchy, in a certain sense. The first system adds machinery that is extra-linguistic. It is machinery for the formulation of proofs, and is not itself formally studied.

The second system is similar, except that the machinery that is added for proof purposes is seen to constitute a formal language of its own. As such, very general results about modal validity can be stated and proved constructively by reasoning

about this enlarged language and its properties. The underlying modal language remains what it was, but we also have a second, richer language for use in proofs.

The third system also enlarges the basic modal language itself, but it does so in a way that increases its expressibility, and our conventional modal language becomes just a fragment of the full language that is used. It turns out that, through the mechanisms expressible in the extended language, we have everything we need for natural tableau systems.

It should be noted that the hierarchical relationship between the three systems we discuss played no role in their creation, their primary uses, or their mathematical applications. There are fundamental important differences between them and how they are used, but these involve areas that we are not concerned with in the present book. We simply present them as systems alternate to our prefixed tableaus, that can be found in the literature, and that could have been used instead of prefixed tableaus to present our main ideas.

7.7.1 Priest Style Tableaus

In his textbook, *An Introduction to Non-Classical Logic, From If to Is*, Priest (2008) discusses a wide range of logics, not just modal, and provides a relatively uniform style of tableau proof system for each. In his style of tableau, each formula is followed by a non-negative integer which can be thought of as ‘naming’ a possible world. Since, at each possible world of a modal model, the classical connectives behave in the usual boolean way, the tableau rules for them are analogs of our rules from Definition 7.1.2. Here are two examples, as representative.

$$\frac{X \wedge Y, i}{\begin{array}{l} X, i \\ Y, i \end{array}} \quad \frac{\neg(X \wedge Y), i}{\neg X, i \mid \neg Y, i}$$

Since integers do not carry the accessibility information that our prefixes do, there is a second kind of item allowed in a tableau, of the form irj , where r is a binary relation symbol and i and j are non-negative integers. We can think of it as saying the world that j names is accessible from the world that i names. These irj expressions are not part of the formal modal language, but are part of what we might call the *proof language*. Some modal rules can only be fired if certain expressions of this form already appear on a branch; others add formulas of this form to a branch. Here are the Priest style analogs of our prefixed Necessity and Possibility rules, Definitions 7.1.4 and 7.1.3, written in the format used in his book. In the rightmost two rules below, the possibility cases, the integer j must be *new* to the branch. The first two rules say that if a formula of the appropriate form, shown above the arrow, is already on the branch, and irj is also present, the formula below the arrow can be

added. The second two say that if the formula above the arrow is present, the two items below the line may be added, *with the proviso that j is new*.

$$\begin{array}{cc}
 \Box X, i & \neg \Diamond X, i \\
 \text{---} & \text{---} \\
 irj & irj \\
 \downarrow & \downarrow \\
 X, j & \neg X, j
 \end{array}
 \qquad
 \begin{array}{cc}
 \Diamond X, i & \neg \Box X, i \\
 \downarrow & \downarrow \\
 irj & irj \\
 X, j & \neg X, j
 \end{array}$$

A proof of a formula X is a closed tableau beginning with $\neg X, 0$. A tableau is closed if each branch contains formulas of the forms Z, i and $\neg Z, i$. Example 7.7.1 is a Priest-style proof for the formula $\Box(P \wedge Q) \supset (\Box P \wedge \Box Q)$, which was previously seen in Example 7.1.6. Notice that on the left branch, the first modal rule applied is the one for the negation of a necessitated formula, which introduces a new integer, 1, to the branch and adds $0r1$ as well as $\neg P, 1$. Then $P \wedge Q, 1$ is added using the rule for an unnegated necessitated formula. When the rule is applied, $0r1$ is present, so the rule application is permitted. The right branch is similar. Note that each branch is a separate case, so when 1 is introduced on the right branch, the fact that it already occurs on the left branch is not relevant.

Example 7.7.1 (Priest Tableau)

$$\begin{array}{c}
 \neg[\Box(P \wedge Q) \supset (\Box P \wedge \Box Q)], 0 \\
 \Box(P \wedge Q), 0 \\
 \neg(\Box P \wedge \Box Q), 0 \\
 \swarrow \quad \searrow \\
 \begin{array}{l}
 \neg\Box P, 0 \\
 0r1 \\
 \neg P, 1 \\
 P \wedge Q, 1 \\
 P, 1 \\
 Q, 1
 \end{array}
 \quad
 \begin{array}{l}
 \neg\Box Q, 0 \\
 0r1 \\
 \neg Q, 1 \\
 P \wedge Q, 1 \\
 P, 1 \\
 Q, 1
 \end{array}
 \end{array}$$

Consequence plays an important role in the Priest book. To quote, “the initial list for the tableau comprises $A, 0$, for every premise A (if there are any), and $\neg B, 0$, where B is the conclusion.” A comparison with Definition 7.3.1 and a little thought shows that the premises are being taken as *local* assumptions in our sense.

The tableau rules given so far are the basic ones, and the resulting logic is **K**. Rules for other logics, whose semantics involve reflexivity, symmetry, transitivity, and seriality, are added in quite a direct way. Here they are, with the Greek letter names used in Priest (2008). You might try them on some examples.

Name	Property	Rule	Restriction
ρ	reflexivity	$\downarrow$ iri	i not new
σ	symmetry	irj $\downarrow$ jri	
τ	transitivity	irj jrk $\downarrow$ irk	
η	seriality	$\downarrow$ irj	i not new, j new

7.7.2 Negri Style Tableaus

The title of this part is not quite accurate. The approach that is found in Negri (2005) and in Negri and von Plato (2001) does not actually use tableaus. It uses what are called *sequent calculi*. Sequent based proof systems are common in the literature, and are used by most researchers who study what is called *proof theory*. Loosely, proof theory is the constructive study of properties of logics via the detailed examination of the properties possessed by their proof mechanisms, and these mechanisms are almost always restricted to sequent calculi. Proof theory is not an area we will examine in the present book, so we say no more about it. Except: it is the case that, loosely speaking, a sequent calculus turned upsidedown and run backward is, essentially, a tableau system. So, it is easy to present the Negri style approach as being about tableaus, even though the original sources are written in terms of sequent calculi. If this has been confusing, let it go. What follows is what the Negri style modal sequent systems look like when re-formulated in tableau form.

We have our modal propositional language, as specified in Definition 5.1.1. We use the term *modal formula* for members of this language. Now a set of *variables*, $x_1, x_2, \dots$ is added. We will use $x, y, z, \dots$ informally for these. A binary relation symbol R is added, and xRy is a *relational atom*. In addition, if x is a variable and Z is a modal formula, then $x : Z$ is a *labeled formula*. Informally, one can think of x as naming a possible world, and $x : Z$ as saying that Z is true at that world. We can think of relational atoms and labeled formulas together as making up a formal language, and so methods of study that apply to formal languages can be applied here. For us, the main point is that it is formulas from this language that occur in tableaus.

A Negri-style proof of a modal formula Z is a closed Negri tableau beginning with $x : \neg Z$, where x is an arbitrary variable. A tableau branch is closed if it contains

both $y : X$ and $y : \neg X$ for some variable y and modal formula X ; if X is atomic, the branch is atomically closed. A tableau branch is also atomically closed if it contains xRy and $\neg xRy$. The rules for extending branches (for the logic **K**) are given in Definition 7.7.2.

Definition 7.7.2 (Negri-Style Tableau Rules for K)

$$\text{Propositional Rules} \quad \frac{x : X \wedge Y}{\begin{array}{l} x : X \\ x : Y \end{array}} \quad \frac{x : \neg(X \wedge Y)}{x : \neg X \mid x : \neg Y}$$

Other propositional cases are similar.

$$\text{Basic Necessity Rules} \quad \frac{x : \Box X}{y : X} \quad \frac{x : \neg \Diamond X}{y : \neg X}$$

Possibility Rules If the variable y is new to the branch,

$$\frac{x : \Diamond X}{\begin{array}{l} xRy \\ y : X \end{array}} \quad \frac{x : \neg \Box X}{\begin{array}{l} xRy \\ y : \neg X \end{array}}$$

In the Priest style tableau systems, machinery is added for use in the proof mechanism, such as expressions like $2r3$. In Negri style tableau systems such machinery is itself part of a formal (but enlarged) language. Put briefly, in Priest tableaus the proof machinery is external, but in Negri tableaus it is internal. Proof theoretic methods that have been developed over many years of research can be brought to bear on things in the Negri approach. And further, this can be done in a uniform way, covering a wide range of modal logics. Still, this is not among our concerns here. For further information the reader should consult Negri (2005).

The Priest style tableau given as an example earlier is easily converted into a Negri style proof. At the beginning, x is a new variable.

Example 7.7.3 (Negri Tableau)

$$\begin{array}{c} x : \neg[\Box(P \wedge Q) \supset (\Box P \wedge \Box Q)] \\ x : \Box(P \wedge Q) \\ x : \neg(\Box P \wedge \Box Q) \\ \swarrow \quad \searrow \\ \begin{array}{l} x : \neg \Box P \\ xRy \\ y : \neg P \\ y : (P \wedge Q) \\ y : P \\ y : Q \end{array} \quad \begin{array}{l} x : \neg \Box Q \\ xRy \\ y : \neg Q \\ y : (P \wedge Q) \\ y : P \\ y : Q \end{array} \end{array}$$

Rules for introducing reflexivity, symmetry, and the like are quite obvious, and also essentially the same as for the Priest system.

At around the same time that the present second edition of *First-Order Modal Logic* was under development, Orlandelli (2021) appeared. It, in fact, provides a detailed proof-theoretic formulation of the machinery discussed above, but in place of the prefixed tableaus we have chosen to use, it makes use of a Negri-style labeled sequent calculus. It is discussed how the work presented here can be developed in such a context. This paper is recommended to those readers who want to follow up on the possibilities and possible advantages of such an approach.

7.7.3 Hybrid Logic Tableaus

The Negri approach just discussed enhances the language that is used in proofs, but the basic modal logic remains untouched. Hybrid logic extends the basic language itself, with *nominals*, and with operators to manipulate them. The motivation ultimately traces back to work of Prior on temporal logic. With temporal analogs of modal operators one can talk about things like “always in the future” or “sometime in the past.” What Prior also added to this was machinery to talk about *instants*, for instance “at noon on February 11, 2000.” Nominals are the generalization of instants whereby within a modal language one can talk about specific possible worlds. Here is a quick sketch of the basic ideas. For a thorough general discussion see (Blackburn et al., 2001, Section 7.3) and Areces and ten Cate (2007), and for material directly related to the subject matter of the present book, see Braüner (2011).

Our basic modal language is enhanced (for this section only) by the addition of a distinct set of propositional letters which we will write as $a, b, c, \dots$. These are called *nominals*. The semantics is enhanced with a condition that says that in a model, a nominal must be assigned truth only *at a single possible world*. In this way, nominals can be thought of informally as names for particular worlds, and we will say things like Γ is the world named by a in a model, if Γ is the unique world in which a is true.

In addition to the usual modal operators $\Box$ and $\Diamond$, a hybrid language has operators that relate to the behavior of nominals. There is a range of such operators, giving logics of various strengths, but ones that are always present are the *satisfaction operators*, often called *at operators*. These are written as $@_a$ for each nominal a , and are prefix operators, like $\Box$ and $\Diamond$. Their semantic behavior is: in a model, $@_a X$ is true at every world if and only if X is true at the world named by the nominal a . This greatly enhances the expressibility of a modal logic. For instance, using nominals a and b , the formula $@_a \Diamond b$ is true at every world of a modal model if and only if the world named by b is accessible from the world named by a . That is, the accessibility relation can be internalized using the machinery of nominals.

There are more operators that can be added to the version of hybrid logic that has been presented so far. But already, it is rich enough so that tableau systems for it exist without needing any additional machinery, just like classical logic in Chap. 3. Since the hybrid logic we have sketched includes the standard modal operators, then a hybrid tableau system gives us one for pure modal logic as well. Since no conditions have been placed on the accessibility relation, this would be a tableau system appropriate for **K**. In fact, there are several such hybrid tableau systems. We present one that appears in Brauüner (2011). In it, all entries are satisfaction statements or their negations. That is, of the forms $@_a X$ or $\neg @_a X$. Not all hybrid tableau systems are like this. On the other hand, they are closest to the other tableau systems seen in this book.

A hybrid tableau proof of a formula X , where X can contain hybrid machinery, is a closed tableau beginning with $\neg @_n X$, where n is a nominal not occurring in X . A tableau is closed if each branch contains a contradiction, Z and $\neg Z$ for some Z . Branch Extension Rules, for a hybrid version of **K**, are given in Definition 7.7.4.

Definition 7.7.4 (Hybrid Tableau Rules for K)

$$\textbf{Propositional Rules} \quad \frac{@_a \neg X}{\neg @_a X} \quad \frac{\neg @_a \neg X}{@_a X} \quad \frac{@_a (X \wedge Y)}{@_a X} \quad \frac{\neg @_a (X \wedge Y)}{\neg @_a X \mid \neg @_a Y} \quad \frac{}{@_a Y}$$

Other propositional cases are similar.

$$\textbf{@ Rules} \quad \frac{@_c @_a X}{@_a X} \quad \frac{\neg @_c @_a X}{\neg @_a X} \quad \frac{}{@_a a} \quad \frac{@_a X}{@_a c} \quad \frac{\neg @_a X}{\neg @_a c}$$

provided a occurs on branch

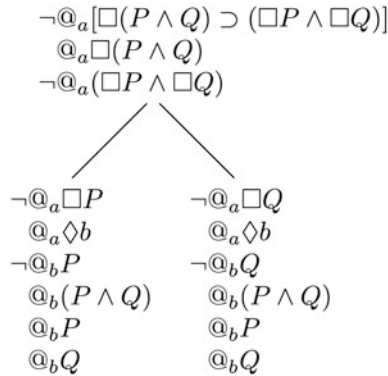
provided c and X occur in the formula being proved

$$\textbf{Basic Necessity Rules} \quad \frac{@_a \Box X}{@_a \Diamond e} \quad \frac{\neg @_a \Diamond X}{\neg @_a \Box X}$$

Possibility Rules If the nominal c is new to the branch,

$$\frac{@_a \Diamond X}{@_a \Diamond c} \quad \frac{\neg @_a \Box X}{@_a \Diamond c}$$

Example 7.7.5 shows a proof using the hybrid tableau system just described. It is, in fact, the same example we have been following, for $\Box(P \wedge Q) \supset (\Box P \wedge \Box Q)$. We take a as a new nominal to start things off.

Example 7.7.5 (Hybrid Example)

The tableau system just given is for the hybrid version of **K**. That is, when used to prove formulas that do not involve any nominals it proves the formulas of **K** (nominals will turn up in the proof, however). But this system can also be used to prove formulas that themselves involve nominals, and so it is richer than the prefixed tableau system or the systems of Priest and Negri. For instance, semantically nominals name *unique* worlds, so if we have $@_i j$, then i and j should name the same world. If we also have $@_i k$ then i and k name the same world. So then k and j should name the same world. We do not have equality available to us directly, but we can conclude that we must also have $@_k j$. Our conclusion is that, if the system is complete, the formula $(@_i j \wedge @_i k) \supset @_k j$ should have a tableau proof. In fact, it does, shown in Example 7.7.6. We have numbered lines for reference purposes.

Example 7.7.6 (Hybrid Example with Nominals Only)

- $\neg @_a[(@_i j \wedge @_i k) \supset @_k j]$ 1.
- $@_a(@_i j \wedge @_i k)$ 2.
- $\neg @_a @_k j$ 3.
- $@_a @_i j$ 4.
- $@_a @_i k$ 5.
- $\neg @_k j$ 6.
- $@_i j$ 7.
- $@_i k$ 8.
- $@_k j$ 9.

In the example, propositional rules take us through line 5. The first two of the @ Rules gives us 6, 7, and 8. Finally, 9 follows from 7 and 8 using the fourth of the @ Rules, where we take a to be i , X to be j , and c to be k . Closure is by 6 and 9.

Finally, a move beyond **K** does not need additional rules; instead very simple axioms will do since nominals can be involved. For instance, reflexivity is axiom-

atized by $i \supset \Diamond i$. But remarkably, irreflexivity is also axiomatizable, by $i \supset \neg \Diamond i$. This cannot be captured by axioms that do not involve nominals. Hybrid logic is a very expressive modal system.

Just as work on the present book edition was being planned, Indrzejczak (2020) appeared. It provides a proof-theoretic development of the approach of this book, but based on hybrid logic. It is a recommended source for those interested in exploring hybrid logic further.

References

- Areces, C., & ten Cate, B. (2007). In Blackburn, Benthem, & Wolter (2007). In P. Blackburn, J. V. Benthem, & F. Wolter (Eds.), (Chap. 14: Hybrid Logics, pp. 821–868). Elsevier.
- Blackburn, P., Benthem, J. V., & Wolter, F. (Eds.). (2007). *Handbook of modal logic*. Studies in Logic and Practical Reasoning. Amsterdam: Elsevier.
- Blackburn, P., de Rijke, M., & Venema, Y. (2001). *Modal logic*. Tracts in Theoretical Computer Science. Cambridge: Cambridge University Press.
- Braüner, T. (2011). *Hybrid logic and its proof-theory*. Applied Logic Series. Dordrecht: Springer.
- D’Agostino, M., Gabbay, D., Hähnle, R., & Posegga, J. (Eds.). (1999). *Handbook of tableau methods*. Dordrecht: Kluwer.
- Fitting, M. (1972). Tableau methods of proof for modal logics. *Notre Dame Journal of Formal Logic*, 13, 237–247.
- Fitting, M. (1983). *Proof methods for modal and intuitionistic logics*. Dordrecht: D. Reidel.
- Fitting, M. (1999). *Introduction* (chap. Introduction, pp. 1–43). In M. D’Agostino, D. Gabbay, R. Hähnle, & J. Posegga. Cambridge: Kluwer.
- Goré, R. (1998). *Tableau methods for modal and temporal logics*. In M. D’Agostino, D. Gabbay, R. Hähnle, & J. Posegga (1999).
- Henkin, L. (1949). The completeness of the first-order functional calculus. *Journal of Symbolic Logic*, 14, 159–166.
- Indrzejczak, A. (2020). Existence, definedness and definite descriptions in hybrid modal logic. In N. Olivetti, R. Verbrugge, S. Negri, & G. Sandu (Eds.), *Advances in modal logic* (vol. 13, pp. 349–368). London: College Publications.
- Massacci, F. (1998). *Single step tableaux for modal logics: Methodology, computations, algorithms* (Technical Report No. TR-04). Dipartimento di Informatica e Sistemistica, Università di Roma “La Sapienza”.
- Massacci, F. (2000). Single step tableaux for modal logics: Computational properties, complexity and methodology. *Journal of Automated Reasoning*, 24, 319–364.
- Negri, S. (2005). Proof analysis in modal logic. *Journal of Philosophical Logic*, 34, 507–544.
- Negri, S., & von Plato, J. (2001). *Structural proof theory*. Cambridge: Cambridge University Press. (Paperback edition 2008)
- Orlandelli, E. (2021). Labelled calculi for quantified modal logics with definite descriptions. *Journal of Logic and Computation*, 31(3), 923–946.
- Priest, G. (2008). *An introduction to non-classical logic: From if to is*. Cambridge Introductions to Philosophy (2nd edn.). Cambridge: Cambridge University Press. First Edition published 2001.

Part III

First-Order Modal Logic

This is a book on first-order modal logic, and we have finally reached the point where we can start talking about it. Our background material is finished.

Adding quantifier machinery to classical propositional logic yields first-order classical logic, fully formed and ready to go. For modal logic, however, adding quantifiers is far from the end of the story, as we will soon see. But certainly adding quantifiers is the place to start. As it happens, even this step presents complications that do not arise classically. Once some of the basic complexities are brought out semantically, we will turn to proof theory. From here on, tableaux play the central role. We will discuss axiomatics briefly, but not as fully as we did in earlier chapters. But, as usual, we begin with basic syntax.

Chapter 8

Quantified Modal Logic



8.1 First-Order Modal Formulas

We have the same *propositional connectives* that we had in Chap. 5, and the same *modal operators*. To these are added two *quantifiers*, $\forall$ (*for all*, the universal quantifier) and $\exists$ (*there exists*, the existential quantifier). Just as with the modal operators, these turn out to be interdefinable, and sometimes it is convenient to take one as basic and the other as defined.

We assume we have available a countable set of *one place relation symbols* (also called *predicate symbols*), a countable set of *two place relation symbols* disjoint from the first set, a countable set of *three place relation symbols* disjoint from the first two sets, and so on. We also allow *zero place relation symbols*, which can be identified with the propositional letters used when we discussed propositional modal logic. In this way quantified modal logics can be seen as directly extending their propositional versions. But since we have already discussed propositional logics, none of the examples shown from here on will actually involve zero place relation symbols.

An n -place relation symbol is said to be *of arity n* . It is fairly common to see relation symbols explicitly given, say as $\{P_1^1, P_2^1, P_3^1, \dots\}$ for the set of one place relation symbols, and similarly for the others. There is little point to this, since we will never see such details again. We will be informal in our notation and use P , R , or something similar for a relation symbol, with its arity determined from context. All that matters is that we have a distinct countable set of arity n for each n , where *countable* means its members can be listed in first, second, third, etc. order if needed.

We also assume we have a countable set of *variables*. Here too we will be informal in our notation, and write x , y , z , and the like, for variables. Quantified logics often also contain constant and function symbols. We do not incorporate them now, though we will bring them in when we get to Part V.

Definition 8.1.1 (Atomic Formula) An *atomic formula* is any expression of the form $R(x_1, x_2, \dots, x_n)$, where R is an n -place relation symbol and $x_1, x_2, \dots, x_n$ are variables.

The propositional letters of Chap. 5 were stand-ins for propositions which were not further analyzed. In a first-order logic, atomic formulas have more structure to them. Think of a relation symbol as standing for a relation, such as *is-the-brother-of* (two-place) or *is-a-number-between* (three-place). An atomic formula like $R(x, y)$ is supposed to mean: the relation that R stands for holds of the things that are the values of x and y . This will be given a precise meaning when semantics is discussed.

In previous chapters we followed a convention of using uppercase Latin letters, $X, Y, A, B, \dots$, to stand for arbitrary formulas of propositional logics. But now we have reached the complexities of first-order logics and, as we noted above, uppercase Latin letters will stand for relation symbols, and are specific parts of *atomic* formulas. To keep confusion down, from here on when we are discussing arbitrary first-order formulas we will use Φ, Ψ , and other uppercase Greek letters. Uppercase Latin letters only stand for relation symbols of our language from now on. Just letting you know.

Next we define the notion of *first-order formula*, and simultaneously we define the notion of *free variable occurrence*. Anticipating notation that we are about to define formally, we will see that $((\forall x)P(x) \supset Q(x))$ is a formula. In it, the occurrence of x in $P(x)$ is within the scope of the quantifier $(\forall x)$, and is not counted as free while the occurrence in $Q(x)$ is free. Think of the variable x in $P(x)$ or $Q(x)$ as if it were the pronoun “it”. Then $Q(x)$ says “it is Q ”. We are assuming what thing it is has been made clear by context. In $(\forall x)P(x)$ we are saying “everything is such that *it* is P .” Here the “it” is not deictic but anaphoric on the indefinite pronoun “everything”.

Definition 8.1.2 (First-Order Modal Formulas) The sets of *first-order formulas* and *free variable occurrences* are determined by the following rules.

1. Every atomic formula is a formula; every occurrence of a variable in an atomic formula is a free occurrence.
2. If Φ is a formula, so is $\neg\Phi$; the free variable occurrences of $\neg\Phi$ are those of Φ .
3. If Φ and Ψ are formulas and $\circ$ is a binary connective, $(\Phi \circ \Psi)$ is a formula; the free variable occurrences of $(\Phi \circ \Psi)$ are those of Φ together with those of Ψ .
4. If Φ is a formula, so are $\Box\Phi$ and $\Diamond\Phi$; the free variable occurrences of $\Box\Phi$ and of $\Diamond\Phi$ are those of Φ .
5. If Φ is a formula and v is a variable, both $(\forall v)\Phi$ and $(\exists v)\Phi$ are formulas; the free variable occurrences of $(\forall v)\Phi$ and of $(\exists v)\Phi$ are those of Φ , *except* for occurrences of v .

Any variable occurrences in a formula that are not free are said to be *bound*.

Example 8.1.3 Consider $(\exists y)((\forall x)P(x, y) \supset \Box Q(x, y))$, where P and Q are both two-place relation symbols. We sketch why this is a formula, and determine which

variable occurrences are free. To help make things clear, we display free variable occurrences in formulas using bold-face.

Both $P(\mathbf{x}, \mathbf{y})$ and $Q(\mathbf{x}, \mathbf{y})$ are atomic formulas, hence both are formulas, and all variable occurrences are free.

Since $Q(\mathbf{x}, \mathbf{y})$ is a formula, so is $\Box Q(\mathbf{x}, \mathbf{y})$, and it has the same free variable occurrences.

Since $P(\mathbf{x}, \mathbf{y})$ is a formula, so is $(\forall x)P(x, \mathbf{y})$, and the free variable occurrences are those of $P(\mathbf{x}, \mathbf{y})$, except for occurrences of x , hence only the occurrence of $\mathbf{y}$ is free.

Now $((\forall x)P(x, \mathbf{y}) \supset \Box Q(\mathbf{x}, \mathbf{y}))$ is a formula, with free variable occurrences as indicated.

Finally, $(\exists y)((\forall x)P(x, y) \supset \Box Q(\mathbf{x}, y))$ is a formula, and its free variable occurrences are those of $((\forall x)P(x, y) \supset \Box Q(\mathbf{x}, y))$, except for occurrences of y . Thus only the final occurrence of $\mathbf{x}$ is free.

Facts about formulas are, as in previous chapters, often proved by using Complete Induction on formula degree. The following extends Definition 5.1.2 to cover quantifiers.

Definition 8.1.4 (Degree) The degree of a first-order modal formula is the total number of occurrences of $\neg, \wedge, \vee, \supset, \Box, \Diamond, \forall, \exists$ in it.

We will often need to replace occurrences of a variable in a formula with occurrences of a different variable. But we will only do this for *free* occurrences. It is possible to introduce detailed notation for substitution, but we will be somewhat informal, following a practice that is quite common.

Definition 8.1.5 (Substitution) When we write $\Phi(x)$, we are indicating that this is a formula in which the variable x may have free occurrences. If, subsequently, we write $\Phi(y)$, we mean the formula that results from $\Phi(x)$ when all free occurrences of x have been replaced by occurrences of y .

For instance, consider the formula from Example 8.1.3. If we say this is $\Phi(x)$, and later write $\Phi(z)$, where z is a new variable, we have substituted occurrences of z for the *free* occurrences of x , but not for the bound occurrences. The result, then, is the formula $(\exists y)((\forall x)P(x, y) \supset \Box Q(z, y))$. We have replaced one free variable by another. In some sense the structure of the formula has not changed; we have just shifted unimportant details of notation.

One must be careful here. It is not the case that any free variable is as good as any other. If we continue with the same $\Phi(x)$, but use y instead of z , we get $\Phi(y)$ and this is $(\exists y)((\forall x)P(x, y) \supset \Box Q(y, y))$. This does not have the same structure as $(\exists y)((\forall x)P(x, y) \supset \Box Q(z, y))$ at all. The z we introduced earlier occurred free in the formula, but the occurrence of y we now introduce turns out to be bound. We don't want this since it alters meaning. We will avoid such problems when working with tableaux because we will introduce a new set of free variables, *parameters*, that cannot be bound. Here is how we guard against such problems when we don't have a device like parameters available.

Definition 8.1.6 (Substitutability) We say a free variable z is *substitutable* for x in $\Phi(x)$ provided no free occurrence of x in $\Phi(x)$ is in a subformula beginning with $(\forall z)$ or $(\exists z)$. That is, no free occurrence of x in $\Phi(x)$ is within the scope of $(\forall z)$ or $(\exists z)$.

If z is substitutable for x in $\Phi(x)$, then indeed $\Phi(x)$ and $\Phi(z)$ say essentially the same things, except that in discussing $\Phi(z)$ we must say “ z ” where in discussing $\Phi(x)$ we would say “ x .”

When models are defined later on, we will see that a formula without free variables is simply true or false at a world of a model. But for a formula with free variables, additional information must be supplied before truth can be determined, namely values for the free variables. This suggests that formulas without free variables will play a special role.

Definition 8.1.7 (Sentence) We call a formula with no free variable occurrences a *sentence* or a *closed formula*.

Finally, we will continue our earlier practice of informally omitting outer parentheses from formulas, and using differently shaped parentheses.

Exercises

Exercise 8.1.1 Verify that the following are formulas, and determine which are the free variable occurrences. Also, determine which of the following are sentences.

1. $((\forall x)\Diamond P(x, y) \supset (\exists y)\Box Q(x, y))$
2. $(\exists x)(\Box P(x) \supset (\forall x)\Box P(x))$
3. $(\forall x)((\exists y)R(x, y) \supset R(y, x))$

8.2 An Informal Introduction

Let’s look at the kinds of things we can say with first-order formulas before quantifiers are taken into account.

Contrast the formula

$$F(x) \tag{8.1}$$

with the formula

$$\Box F(x). \tag{8.2}$$

(8.1) says that an object x has the property F —in the actual world, of course. Equation (8.2) says that an object x has the property F necessarily, i.e., in every

possible world.¹ The x that occurs in both (8.1) and (8.2) is a free variable, not a name or a description or a function name of any sort. Our formal machinery will include a mechanism for assigning values—objects—to free variables. Informally, we will just refer to the object x . It is important to keep in mind that it is the object itself—the value of x —that we are speaking about, and not the variable.

The contrast between (8.1) and (8.2) is just the contrast, for example, between saying that *God exists* and saying that *God necessarily exists*. If F stands for “exists” and x has God as its value, then (8.1) says that God exists in this world. It does not, of course, say that he exists in every possible world. And if he did not exist in every possible world, then his existence would only be contingent. This is a matter of considerable importance in classical theology. For, it has been claimed that God differs from other things in that it is part of God’s essence that he exists. This is to say that it is an essential property of him that he exists. And, indeed, (8.2) can be read as *F is an essential property of x* .

On the crudest version of the *Ontological Argument*, the claim that it is part of God’s essence that he exists is interpreted as (8.2), and so the fact of his existence, (8.1), follows immediately *via* $\Box F(x) \supset F(x)$ (which is valid in a frame at least as strong as **T**).

But there are more subtle ways of understanding the claim that it is part of God’s essence that he exists. Consider

$$F(x) \supset \Box F(x), \quad (8.3)$$

which says that x has F necessarily if *it has F at all*. This is not generally true: just because x is F in the actual world, it does not follow that it is F in every possible world. But something special happens for the case of God’s existence. (8.2) categorically asserts that x is F in every possible world; (8.3) asserts this only hypothetically, i.e., on the condition that x is F in this world. So, for the case of God’s existence, the contrast is between saying that *God necessarily exists* and saying that *God necessarily exists if he exists*. The distinction is important. It has been a matter of some concern among Descartes’ commentators which of these two is the proper conclusion of the version of the *Ontological Argument* he presents in Meditation V. For if it is (8.3), then, of course, God’s existence has not been established.

We can make yet finer discriminations here if we contrast (8.3) with

$$\Box(F(x) \supset \Box F(x)). \quad (8.4)$$

(8.4) contains one $\Box$ nested inside another. This is where informally interpreting modal formulas becomes difficult; it is also where possible world semantics is most helpful. Equation (8.4) says that (8.3) is necessary, i.e., that it holds in every possible

¹ Understood, of course, as “in every *accessible* possible world.” We will generally suppress “accessible” in our informal discussion.

world. Now, (8.3) says that the following holds in the actual world:

Either x isn't F or x is F in every possible world.

So, (8.4) says that this holds in *any world*.

Returning to our example, the contrast is between saying that *God necessarily exists if he exists* and saying that *Necessarily, God necessarily exists if he exists*. Once again, one might balk at the idea that it is only a contingent property of him that he necessarily exists if he exists.

Our first-order language enables us to make very subtle distinctions that greatly enrich our understanding of traditional modal claims. In this section, we have focused on open formulas, i.e., formulas that contain free variables. In the next section, we will look at quantified formulas.

Exercises

Exercise 8.2.1 Consider the claim

It is necessary that an omniscient being is essentially omniscient

Does this require that an omniscient being exist in more than one possible world (if it exists at any)? Suppose it is possible that there is an omniscient being. Does it follow that one exists?

8.3 Necessity De Re and De Dicto

The sentence

Everything is necessarily F (8.5)

contains an ambiguous construction that has caused much confusion in the history of modal logic. We can express these two readings of (8.5) a bit more clearly as

It is a necessary truth that everything is F (8.6)

and

Each thing is such that it has F necessarily. (8.7)

The medieval logicians were aware of these two interpretations. Equation (8.6) expresses that a proposition [*dictum*] is necessary, and so it is an example of what

they called *necessity de dicto*. Equation (8.7) expresses that a thing [*res*] has a property necessarily. It is an example of *necessity de re*.

We find the same *de re/de dicto* ambiguity in

$$\text{Something necessarily exists.} \quad (8.8)$$

On the one hand, (8.8) could mean

$$\text{It is necessarily true that something exists.} \quad (8.9)$$

This is the *de dicto* reading, and it is true in our semantics simply because, following standard procedure, quantifier domains are taken to be non-empty. On the other hand, the *de re* reading of (8.8),

$$\text{Something has the property of existence essentially,} \quad (8.10)$$

is quite controversial. (8.10) commits us to a necessary existent—perhaps God, but, in any event, an object that exists in every possible world.

Here is an example familiar to those who have followed the philosophical literature on modal logic.²

$$\text{The number of planets is necessarily even.} \quad (8.11)$$

On the *de dicto* reading, (8.11) says that a certain proposition is necessary, namely, the proposition *that the number of planets is even*. And on this *de dicto* interpretation, (8.11) is clearly false, for surely it is only a contingent fact that the number of planets is even. Had history been different, there could have been 9 planets in our solar system. But the *de re* reading of (8.11) says, of the number of planets, that number is necessarily even. And as a matter of fact, the number of planets is 8, and it is an essential property of the number 8 that it is even.

It turns out that the *de re/de dicto* distinction is readily expressible in our first-order modal language as a *scope* distinction. We take the *de dicto* (8.6) to be

$$\Box(\forall x)F(x) \quad (8.12)$$

which asserts, of the statement $(\forall x)F(x)$, that it is necessary. And we take the *de re* (8.7) to be

$$(\forall x)\Box F(x). \quad (8.13)$$

² This example was, famously, developed by Quine. Pluto was deemed a planet when Quine published his example so there were, as understood at the time, nine planets. We have changed the example to accord with the current count.

which asserts, of each thing, that it has F necessarily. Then (8.12) says, *In every possible world, everything is F* , and (8.13) says, *Everything is, in every possible world, F* . We take the *de dicto* and *de re* readings of

Something is necessarily F

to be, respectively,

$$\Box(\exists x)F(x) \quad (8.14)$$

and

$$(\exists x)\Box F(x). \quad (8.15)$$

(8.14) says, *In every possible world, something is F* . Equation (8.15) says, *Something is, in every possible world, F* .

If we regard a modal operator as a kind of quantifier over possible worlds, then the *de re/de dicto* distinction corresponds to a permutation of two types of quantifiers. This way of viewing the matter makes the difference readily apparent. For example, (8.15) requires that there be some one thing (at least)—let's call it a —which is, in every possible world, F . So, a must be F in world Γ , a must be F in world Δ , and so on. Equation (8.14) requires only that in each world there be something that is F , but it does not require that it be the same thing in each world.

The sort of ambiguity exhibited in (8.11) rarely arises in classical logic, but it can—most famously when the item picked out by a definite description fails to exist. Whitehead and Russell (1925–1927) introduced notation as part of their theory of definite descriptions to mark just such a distinction of scope, similar to the scope indicated by quantifier placement. In Part V we will formally introduce a related notation we call *predicate abstraction* that will enable us to mark this distinction explicitly in our modal language. Anticipating a bit, where t is a singular term (say, “the number of planets”) we will distinguish the *de dicto*

$$\Box\langle\lambda x.x \text{ is even}\rangle(t), \quad (8.16)$$

which says, “It is necessary *that* t is even,” from the *de re*

$$\langle\lambda x.\Box(x \text{ is even})\rangle(t), \quad (8.17)$$

which says, “ t is such that it is necessary *of it* that it is even.”

The *de dicto* reading (8.16) is false. As we noted, it is a contingent, not a necessary, fact about the universe that there are exactly 8 planets. The expression “the number of planets” picks out a number via one of its contingent properties. If $\Box$ is to be sensitive to the quality of the truth of a proposition in its scope, then it must be sensitive as well to differences in the quality of terms designating objects—that is, it will be sensitive as to whether the object is picked out by an essential

property or by a contingent one. Since “the number of planets” picks out a number by means of one of its contingent properties, it can pick out different numbers in different possible worlds. In the actual world, that number is 8, but there might have been 9 planets, so in another possible world that number could be 9. The proposition *that the number of planets is even* therefore can come out true in some worlds and false in other. It is not necessarily true, and (8.16) is false.

But the *de re* (8.17) says, “The number of planets is, in every possible world, even.” This is true. There are exactly 8 planets, and so “the number of planets” designates the number 8. Unlike the *de dicto* case, the designation of the term has been fixed in the actual world as 8. And that number, in every possible world, is even.

Note that there is no *de re/de dicto* distinction for an open formula like

$$\Box(x \text{ is even}). \quad (8.18)$$

(8.18) is true of a given object x iff in every possible world that object is even. So (8.18) is true of the number 8; it is false of the number 9. The mode of specification of the object that is the value of x is irrelevant to the truth value of the open formula; the open formula is true of *it*, or not, as the case may be.

Unlike free variables, singular terms—proper names, definite descriptions, function names—require considerable care in modal contexts. To avoid difficulties and confusions, we have chosen to introduce our first-order modal language first without constant or function names, and reserve for later chapters the special problems introduced by them. Once we have predicate abstract notation available, we will be able to investigate the *de re/de dicto* distinction in more detail. That discussion will be found in Chap. 13.

In the next section we will show how the failure to observe the *de re/de dicto* distinction has led to serious errors about the very possibility of coherently doing quantified modal logic.

Exercises

Exercise 8.3.1 Interpreting $\Box$ as *At all future times*, show informally:

1. $\Box(\forall x)F(x)$ fails to imply $(\forall x)\Box F(x)$
2. $(\forall x)\Box F(x)$ fails to imply $\Box(\forall x)F(x)$

Exercise 8.3.2 Does the sentence “Some things are necessarily F” mask a *de re/de dicto* distinction? If so, what are the two readings of the sentence?

8.4 Is Quantified Modal Logic Possible?

For much of the latter half of the twentieth century, there was considerable antipathy toward the development of modal logic in certain quarters. Many of the philosophical objectors found their inspiration in the work of W. V. O. Quine, who as early as Quine (1943) expressed doubts about the coherence of the project. We will find that one of the main sources of these doubts rests on the failure to carefully observe the distinction between *de re* and *de dicto* readings of necessity.

Quantified modal logic makes intelligible the idea that objects themselves, irrespective of how we speak about them, have properties necessarily or contingently.³ Our semantics for modal logic will not require that any particular object have any particular substantial property either accidentally or essentially, but only that it makes sense to speak this way. We leave it to metaphysicians to fill in their details.

Quine did not believe that quantified modal logic can be done coherently because he takes to be a feature of reality what is actually a feature of language. He says, for example, “Being necessarily or possibly thus and so is in general not a trait of the object concerned, but depends on the manner of referring to the object.” (Quine, 1961, p. 148) It is instructive to go over Quine’s reasons for holding this view.

Quine (1953) set up the problem by identifying three interpretations of $\Box$ on which the modality is progressively more deeply involved in our world outlook. On Grade 1, the least problematic level of involvement, $\Box$ is taken to be a metalinguistic predicate that attaches to a name of a sentence, in the same way as the Tarski reading of “is true.” To say that a sentence is necessarily true is no more than to say it is a theorem (of a formal system reasonably close to logic—perhaps including set theory), and the distinction between theorems and nontheorems is clear. But on this reading, there can be no iteration of modal operators, and as a result there is no need for a specifically modal *logic*. There is no interpretation for the many propositional modal logics we studied earlier in this book.

On Grade 2, which is the interpretation we relied on for our discussion of propositional modal logic, $\Box$ is an operator like $\neg$ that attaches to *closed* formulas, that is, to sentences. But there is an important difference between the logical operators $\Box$ and $\neg$. When P and Q have the same truth value, $\neg P$ and $\neg Q$ also have the same truth value, but $\Box P$ and $\Box Q$ need not. For example, although the two sentences “ $8 > 7$ ” and “The number of planets > 7 ” are both true, only the former is necessarily true. This should not surprise us. Modal logic differs from classical logic in its sensitivity to the quality of a statement’s truth, so one would hardly expect $\Box$ to be indifferent to sentences merely happening to have the same truth value.

³ Quine (1953) defines “Aristotelian Essentialism” as “the doctrine that some of the attributes of a thing (quite independently of the language in which the thing is referred to, if at all) may be essential to the thing, and others accidental.” (p. 173–4) This is just quantified modal logic. There is nothing essentially Aristotelian about any of it! However, one begins to look a bit Aristotelian if one holds that there *are* essential properties, and even more so if one holds that there are *non-trivial* essential properties.

At Grade 3, $\Box$ is allowed to attach to *open* formulas, as in $\Box(x > 7)$. This is the level needed to combine modality with quantifiers, for we need to say such things as “Something is such that *it* is necessarily greater than 7.” And it is in the passage from Grade 2 to Grade 3 involvement that Quine finds his problems.⁴

Quine found the contrast between the two sentences,

$$8 > 7$$

and

$$\text{The number of planets} > 7$$

puzzling. The first is necessary, but the second is not. Why is this so? We are speaking about the same thing each time, since the number of the planets is 8. The only difference is in the way in which the thing is picked out. It appears that whether or not the claim is necessary depends, not on the thing talked about, but on the way in which it is talked about. And if so, Quine argued, there can be no clear understanding of whether an open sentence like

$$x > 7$$

is necessarily true or not, for the terms “8” and “the number of planets” on which our intuitions about necessity relied are no longer available.

Here is another example of the phenomenon that puzzled Quine, this time involving the notion of *belief*. Although Dr. Jekyll and Mr. Hyde are one and the same person (so the story says), we can very well understand how Robert might believe that Dr. Jekyll is a good citizen and yet not believe that Mr. Hyde is a good citizen. The sentence

Dr. Jekyll is believed by Robert to be a good citizen

is true, but the sentence

Mr. Hyde is believed by Robert to be a good citizen

is false. But then it is a mystery what the open sentence

x is believed by Robert to be a good citizen

is true *of*. Which individual is this? Dr. Jekyll? Mr. Hyde? Which individual this is appears to depend upon how he is specified, not on the individual himself. The

⁴ Of course, Quine had other problems with the connection between necessity and analyticity.

quantified statement “*Someone* is such that *he* is believed by Robert to be a good citizen” seems totally uninterpretable.

Grade 2 involvement appears to imply that the behavior of $\Box$ depends on the way things are picked out; Grade 3 involvement requires that $\Box$ be independent of the way things are picked out. Quine argued that we cannot have it both ways. This difference between Grade 2 and Grade 3 involvement looks very much like the distinction between necessity *de dicto* and necessity *de re*. But it isn’t, and it is important we see that it isn’t. If one assumes there is only *de dicto* necessity at Grade 2, which is surreptitiously what Quine did, then one will have severe problems with *de re* necessity, as happened. If one starts out by assuming that necessity has to do with language and not things, then one will certainly run into problems interpreting modal logic as having to do with things and not language.

As we have pointed out on a number of occasions, the complications for names, descriptions and quantifiers indicates no special problem in interpreting open sentences. In particular, it does not show that necessity is more closely connected with how we specify objects than with the objects themselves. One final example using a temporal interpretation *always* for $\Box$ should make this clear. Contrast the two sentences:

The U.S. President will always be a Democrat. (8.19)

Joe Biden will always be a Democrat. (8.20)

Suppose these are asserted in 2022, when these words were written. And suppose, for simplicity, that Joe Biden never changes his party affiliation. Understood *de re*, then, (8.19) and (8.20) are both true. For that man, Joe Biden or the President of the United States—it does not matter how we specify him—will forever be a Democrat. Understood *de dicto*, however, (8.20) is true, but (8.19) is not—for it is highly unlikely that the Democratic party will have a lock on the Presidency forever. Does the difference in truth value show that *temporality* has more to do with how an object is specified than with the object itself? Hardly. It depends on the fact that the Presidency will be changing hands, and Joe Biden only temporarily holds that office. In the world of 2022, the two coincide; but in later worlds, they don’t.⁵

Exercises

Exercise 8.4.1 Suppose necessity were intrinsically related to the way in which we pick things out. Then we would have to locate the *de re/de dicto* distinction in

⁵ There is an alternative reading of a singular term like “the President of the United States” which takes it to refer to an *intensional entity*. The entity would consist partly of George Washington, partly of Thomas Jefferson, ..., that is, of each of the individuals who occupied the office when they did. See Hughes and Cresswell (1968) for a discussion of this idea.

differences in the way designators designate. Discuss how this distinction might be drawn.

8.5 What the Quantifiers Quantify Over

In first-order modal logic, we are concerned with the logical interaction of the modal operators $\Box$ and $\Diamond$ with the first-order quantifiers $\forall$ and $\exists$. From the perspective of possible world semantics, this is the interaction of two types of quantifiers: quantifiers that range over possible worlds, and quantifiers that range over the objects in those worlds. This interaction leads to complications. In classical logic, Universal Instantiation,

$$(\forall x)\Phi(x) \supset \Phi(x) \quad (8.21)$$

is valid. But the validity of (8.21) in a modal context depends on which particular possible world semantics we choose, i.e., on what we take our quantifiers to quantify over.

In the language of possible world semantics, the formula

$$\Box\Phi(x) \supset \Phi(x) \quad (8.22)$$

says that *If x is Φ in every possible world, then x is Φ in the present possible world.* What is this x ? Whatever it is, we will suppose it to be something that occurs in at least one of the possible worlds in our model.

The semantics we use does not require that objects exist in more than one possible world, but it does not forbid it. There is a fairly well-known alternative semantics for first-order modal logic, most vividly put forward by Lewis (1968), which denies that objects can exist in more than one possible world. Lewis takes objects to be “worldbound.” An object in one world, however, can have a *counterpart* in another possible world. This will be the object in the other world that is (roughly) most similar to the object in this one. (An object will be its own counterpart in any given world.) On the counterpart interpretation, (8.22) says *If in every accessible possible world x ’s counterpart is Φ , then x is Φ .* The main technical problem with counterpart theory is that the *being-a-counterpart-of* relation is, in general, neither symmetric nor transitive, so no natural logic of equality is forthcoming. Counterpart semantics is not what we present here.

What does it mean for an object to exist in more than one possible world? Here is an example. An assassination attempt was made upon Ronald Reagan’s life. He was hit by the bullet, but only injured. Eventually he recovered from his wounds and continued on as President of the United States for many years. But he was almost killed by that bullet; and he could have been killed. This invites us to consider a counterfactual situation in which that man, Ronald Reagan, was indeed killed by that assassin. We are not considering a situation in which a person just like him was

killed by that assassin. We are considering a situation in which that very person, Ronald Reagan, was killed by the assassin. We are imagining the very same person in a counterfactual situation. Since what we mean by a possible world is often just such a counterfactual situation, we are imagining an object—Ronald Reagan—to exist in more than one possible world.

But we have only argued for the coherence of allowing objects to exist in more than one possible world. We have not argued that *all* objects do, let alone that *any* of them do. This is actually a choice to be made in setting up modal models: Since a modal model contains many possible worlds, we need to decide whether the domain of discourse should be fixed for the whole of a modal model, or be allowed to vary from world to world, within the model. Taking the domain to be fixed for the whole model provides the simplest formal semantics. We refer to this as *constant domain* semantics. Allowing modal models to have world-dependent domains gives the greatest flexibility. We refer to this as *varying domain* semantics. In constant domain semantics, the domain of each possible world is the same as every other; in varying domain semantics, the domains of different possible worlds need not coincide, or even overlap. In either case, what we call the *domain of the model* is the union of the domains of the worlds in the model. (For an interesting application of constant domain modal models in mathematics see Smullyan and Fitting (1996), where they provide appropriate machinery for establishing various independence results in set theory.)

There is a certain amount of flexibility available in creating modal models. Certain things are matters of taste or underlying philosophical ideas, and are not dictates of our logic. For instance, we take the domain of the model to be the union of the domains of all the possible worlds. Then, for us, a possible existent is an existent in some possible world. But one could have allowed the domain of the model to be a set that properly includes the union of the domains of the possible worlds. Then one could have possible existents—objects about which we can talk—that don't actually exist in any possible world. This is a variant of what is called *free logic*, and we will have some more to say about it in Sect. 8.8. We have chosen not to take this route. Others may do differently. Models must be coherent—the pieces must fit together usefully. But there can be some variation.

As we said, we are assuming that free variables have as values objects in the domain of the model, not necessarily in the domain of the world we are in. If we suppose we are dealing with constant domain semantics, then (8.21) holds; more generally,

$$(\forall x)\Phi(x) \supset \Phi(y) \tag{8.23}$$

is logically valid. Whenever we speak about an object, that object exists in at least one possible world. In constant domain semantics, what exists at one world exists at all. So any value assigned to y will be in the range of the quantifier $\forall x$. This is analogous to classical logic.

In varying domain semantics, however, the situation is a bit different. No longer does (8.23) always hold. Just because everything in this world is Φ it doesn't follow

that y is Φ , because y might exist only at worlds other than this one, and so not be in the range of the quantifier $(\forall x)$, which we take as ranging over what exists “here.” This is quite unclassical.

There are, accordingly, two very different ways it could happen that x is Φ at a world Γ :

1. x is Φ at Γ and x is in Γ
2. x is Φ at Γ but x need not be in Γ

In the classical situation, we speak of something only if it is in the domain of the model, and therefore in the domain of the quantifier. In the modal situation, however, we want to speak about things that do not exist but could, like Pegasus or the golden mountain or additional planets in the solar system. This is where we have a choice. One solution is to open up the domain of the actual world to all possible objects and keep the classical quantifier rules like Universal Instantiation intact: this is what has sometimes been called *possibilist* quantification. The other solution is to allow the domain of the actual world to differ from the domain of other possible worlds and abandon classical quantifier rules like Universal Instantiation: this is *actualist* quantification. The possibilist quantifier is evaluated for every element of the model $\mathcal{M}$; the actualist quantifier is evaluated at a world Γ only for elements of the domain of the world Γ . Possibilist quantification and actualist quantification correspond to constant domain and varying domain models, respectively. The connection is clear. Constant domain semantics models our intuitions about modality most naturally if we take the domain to consist of possible existents, not just actual ones, for otherwise we would be required to treat every existent as a necessary existent.

Exercises

Exercise 8.5.1 Discuss informally how constant domain semantics will differ from varying domain semantics when the modality is given a temporal interpretation. Which is more natural in the temporal reading?

Exercise 8.5.2 “In constant domain semantics, possible objects exist, so there is no distinction between what is possible and what is actual.” Discuss this claim.

Exercise 8.5.3 “In varying domain semantics, an object can have properties in a world even though it does not exist at that world.” Write an essay either defending or criticizing this claim.

8.6 Constant Domain Models

We begin our formal treatment of semantics for quantifiers with *constant domain* models—*possibilist quantification*. In these models, the domain of quantification is

the same from world to world. Semantically this is somewhat simpler than allowing the domain to vary, and pedagogically it is easier to explain as a first approach. What we present is essentially from Kripke (1963), with some modification of notation. We begin by enhancing the notion of a *frame*, from Definition 5.2.1. (The following was called an *augmented frame* in the first edition of this book, but the term *skeleton* has become rather standard.) Exactly as with propositional modal logic, conditions such as reflexivity, transitivity, and so on, can be placed on accessibility relations. The role such conditions play now is the same as it was propositionally. The action here is with quantifiers and the domains they quantify over. We have little new to say about accessibility here.

Definition 8.6.1 (Skeleton) A structure $\langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ is a *constant domain skeleton* if $\langle \mathcal{G}, \mathcal{R} \rangle$ is a frame and $\mathcal{D}$ is a non-empty set, called the *domain* of the skeleton. If $\langle \mathcal{G}, \mathcal{R} \rangle$ is an $\mathbf{L}$ frame, for a modal logic $\mathbf{L}$, then $\langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ is an $\mathbf{L}$ skeleton.

The domain of a constant domain skeleton is the set of things over which quantifiers can range. It is the same, no matter at what world.

To turn a propositional frame into a model, all we had to say was which propositional letters were true at which worlds. The analog of propositional letters now is atomic formulas, and they have a structure that must be taken into account. More specifically, they involve *relation symbols*, which should stand for relations. But since more than one possible world can be involved, we should say which relation each relation symbol represents, at *each* of the worlds. The following will eventually be extended, beginning with Definition 14.2.1, to cover machinery that has not been introduced yet.

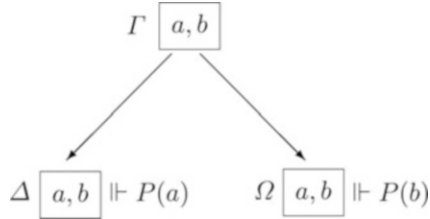
Definition 8.6.2 (Interpretation) $\mathcal{I}$ is an *interpretation* in a constant domain skeleton $\langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ if $\mathcal{I}$ assigns to each n -place relation symbol R and to each possible world $\Gamma \in \mathcal{G}$, some n -place relation on the domain $\mathcal{D}$ of the frame. Thus, $\mathcal{I}(R, \Gamma)$ is an n -place relation on $\mathcal{D}$, and so each n -tuple $\langle d_1, d_2, \dots, d_n \rangle$ of members of $\mathcal{D}$ either is in the relation $\mathcal{I}(R, \Gamma)$ or is not. If $\langle d_1, d_2, \dots, d_n \rangle$ is in the relation $\mathcal{I}(R, \Gamma)$ we will write $\langle d_1, d_2, \dots, d_n \rangle \in \mathcal{I}(R, \Gamma)$, following the standard mathematical practice of thinking of an n -place relation as a set of n -tuples.

Definition 8.6.3 (Model) A constant domain *first-order model* is a structure $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ where $\langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ is a constant domain skeleton and $\mathcal{I}$ is an interpretation in it. By the *domain of the model* $\mathcal{M}$ we mean the domain of its skeleton, $\mathcal{D}$. We say $\mathcal{M}$ is a constant domain first-order model for a modal logic $\mathbf{L}$ if the skeleton $\langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ is an $\mathbf{L}$ skeleton.

This definition will be extended later, beginning with Definition 14.2.2.

Example 8.6.4 Here is our first example of a constant domain first-order model. Let $\mathcal{G}$ consist of three possible worlds, Γ , Δ , and Ω , with $\Gamma \mathcal{R} \Delta$, $\Gamma \mathcal{R} \Omega$, and $\mathcal{R}$ holding in no other cases. Let $\mathcal{D} = \{a, b\}$. Let P be a one-place relation symbol. It is the only relation symbol we are interested in for now, so we will specify an interpretation only for it, and leave things unspecified for any other relation symbols. Now, let $\mathcal{I}(P, \Gamma)$ be the empty set (that is, nothing is in this relation); let $\mathcal{I}(P, \Delta)$

consist of just a ; let $\mathcal{I}(P, \Omega)$ consist of just b . This specifies a constant domain first-order model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$. We represent it schematically as follows.



We have shown the domain $\{a, b\}$ explicitly at each possible world, emphasizing that this is a constant domain model. Notice that we wrote $P(a)$ at the Δ part of the schematic. *This is not a formula!* It is not, since a is a member of the domain associated with Δ , but is not a variable of our formal language. Nonetheless this abuse of notation is handy for specifying models, provided its limitations are understood. Similar remarks apply to $P(b)$ at Ω , of course.

Next we must specify truth in constant domain first-order models. Not surprisingly, this is more complicated than it was in the propositional case, and we hinted at the source of the complications in the example above. We would like to have $(\forall x)P(x)$ be true at a possible world, say Γ , just in case $P(x)$ is true at Γ for all members of the domain $\mathcal{D}$. But, we cannot express this by saying that we want $P(c)$ to be true at Γ for all $c \in \mathcal{D}$, because $P(c)$ will not generally be a formula of our language. A way around this was introduced many years ago by Tarski for classical logic, and is easily adapted to modal models. For $(\forall x)P(x)$ to be true at Γ , we will require that $P(x)$ be true at Γ no matter what member of $\mathcal{D}$ we might have assigned to x as its value. But this means we cannot confine ourselves to the truth of *sentences* at worlds, but instead we must deal with the broader notion of the truth of formulas containing free variables, when values have been assigned to those free variables. We do this using a *valuation* function—akin to what computer scientists call an *environment*. For classical logic this is where *satisfaction* comes in.

Definition 8.6.5 (Valuation) Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a constant domain first-order model. A *valuation* in the model $\mathcal{M}$ is a mapping v that assigns to each free variable x some member $v(x)$ of the domain $\mathcal{D}$ of the model.

What we are about to define is denoted by $\mathcal{M}, \Gamma \Vdash_v \Phi$ where $\mathcal{M}$ is a constant domain first-order model, Γ is a possible world of the model, Φ is a formula, perhaps with free variables, and v is a valuation. Read it as: formula Φ is true at world Γ of model $\mathcal{M}$ with respect to valuation v , where v tells us what values have been assigned to free variables. (Other books may use different notation.) We need one additional piece of technical terminology, then we can go ahead.

Definition 8.6.6 (Variant) Let v and w be two valuations. We say w is an x -variant of v if v and w agree on all variables except possibly the variable x .

Now, here is the fundamental definition.

Definition 8.6.7 (Truth in a Constant Domain Model) Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a constant domain first-order modal model. For each $\Gamma \in \mathcal{G}$ and valuation v in $\mathcal{M}$:

1. If R is an n -place relation symbol, $\mathcal{M}, \Gamma \Vdash_v R(x_1, \dots, x_n)$ provided $\langle v(x_1), \dots, v(x_n) \rangle \in \mathcal{I}(R, \Gamma)$.
2. $\mathcal{M}, \Gamma \Vdash_v \neg\Phi \iff \mathcal{M}, \Gamma \nVdash_v \Phi$.
3. $\mathcal{M}, \Gamma \Vdash_v (\Phi \wedge \Psi) \iff \mathcal{M}, \Gamma \Vdash_v \Phi$ and $\mathcal{M}, \Gamma \Vdash_v \Psi$.
4. $\mathcal{M}, \Gamma \Vdash_v (\Phi \vee \Psi) \iff \mathcal{M}, \Gamma \Vdash_v \Phi$ or $\mathcal{M}, \Gamma \Vdash_v \Psi$.
5. $\mathcal{M}, \Gamma \Vdash_v (\Phi \supset \Psi) \iff \mathcal{M}, \Gamma \nVdash_v \Phi$ or $\mathcal{M}, \Gamma \Vdash_v \Psi$.
6. $\mathcal{M}, \Gamma \Vdash_v \Box\Phi \iff$ for every $\Delta \in \mathcal{G}$, if $\Gamma \mathcal{R} \Delta$ then $\mathcal{M}, \Delta \Vdash_v \Phi$.
7. $\mathcal{M}, \Gamma \Vdash_v \Diamond\Phi \iff$ for some $\Delta \in \mathcal{G}$, $\Gamma \mathcal{R} \Delta$ and $\mathcal{M}, \Delta \Vdash_v \Phi$.
8. $\mathcal{M}, \Gamma \Vdash_v (\forall x)\Phi \iff$ for every x -variant w of v in $\mathcal{M}$, $\mathcal{M}, \Gamma \Vdash_w \Phi$.
9. $\mathcal{M}, \Gamma \Vdash_v (\exists x)\Phi \iff$ for some x -variant w of v in $\mathcal{M}$, $\mathcal{M}, \Gamma \Vdash_w \Phi$.

This definition should be compared with Definition 5.2.3. The last two items are the key additions to the earlier definition. Item 8 says we should take $(\forall x)\Phi$ to be true at Γ , relative to a valuation v , provided Φ is true at Γ no matter what member of the domain $\mathcal{D}$ we assign to x (keeping the values assigned to other variables unchanged, of course). Likewise 9 says a similar thing about $(\exists x)\Phi$.

Proposition 8.6.8 Suppose $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ is a constant domain model, $\Gamma \in \mathcal{G}$, v_1 and v_2 are two valuations in $\mathcal{M}$, and Φ is a formula. If v_1 and v_2 agree on all the free variables of Φ , then

$$\mathcal{M}, \Gamma \Vdash_{v_1} \Phi \iff \mathcal{M}, \Gamma \Vdash_{v_2} \Phi.$$

This Proposition says that if two valuations agree on the free variables actually present in a formula, the behavior of that formula with respect to the two valuations is the same. The proof is an induction argument on the complexity of Φ . It is routine, but complicated, and we omit it. (Or rather, we leave it as an exercise.) In Sect. 8.7 *varying domain* semantics is introduced. Proposition 8.6.8 extends to varying domain models as well, and with essentially the same proof. We remind the reader of Definition 8.1.7, for sentence. The proposition above tells us that truth or falsity of sentences at possible worlds is independent of a choice of valuation.

Definition 8.6.9 (True at Γ) Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a constant domain model with $\Gamma \in \mathcal{G}$. For a *sentence* Φ , if $\mathcal{M}, \Gamma \Vdash_v \Phi$ for some valuation v in $\mathcal{M}$ then $\mathcal{M}, \Gamma \Vdash_v \Phi$ for every valuation v in $\mathcal{M}$ (by Proposition 8.6.8), and conversely. We sometimes abbreviate notation in this situation by writing $\mathcal{M}, \Gamma \Vdash \Phi$, and we simply say Φ is *true at Γ* , but we will only do this for *sentences*.

Now the terminology of Definition 5.4.1 is simply carried over to first-order models. Φ is *valid* in a model if it is true at every world of the model, and so on.

Example 8.6.10 We continue Example 8.6.4, and show $(\forall x)\Diamond P(x) \supset \Diamond(\forall x)P(x)$ is not valid in the model we gave; specifically it fails at Γ . All references to $\mathcal{M}$ are to the model of Example 8.6.4.

Since $(\forall x)\Diamond P(x) \supset \Diamond(\forall x)P(x)$ is a sentence, to show

$$\mathcal{M}, \Gamma \not\models (\forall x)\Diamond P(x) \supset \Diamond(\forall x)P(x)$$

we must show

$$\mathcal{M}, \Gamma \not\models_v (\forall x)\Diamond P(x) \supset \Diamond(\forall x)P(x),$$

where v is some (equivalently any) valuation in $\mathcal{M}$. And to do this, we must show that $\mathcal{M}, \Gamma \models_v (\forall x)\Diamond P(x)$ but $\mathcal{M}, \Gamma \not\models_v \Diamond(\forall x)P(x)$. We do this as follows.

Let w be like v on all variables except that $w(x) = a$. Now, $\mathcal{J}(P, \Delta)$ is the one-place relation that holds of just a , and $w(x) = a$, so $\langle w(x) \rangle \in \mathcal{J}(P, \Delta)$ and by definition we have

$$\mathcal{M}, \Delta \models_w P(x)$$

and consequently

$$\mathcal{M}, \Gamma \models_w \Diamond P(x).$$

In the same way we can show that, if w' is like v on all variables, except that $w'(x) = b$, then

$$\mathcal{M}, \Gamma \models_{w'} \Diamond P(x).$$

Since the domain of $\mathcal{M}$ is $\{a, b\}$, w and w' are all the x -variants of v that there are (indeed, one of them must actually be v itself). Then by item 8 of Definition 8.6.7,

$$\mathcal{M}, \Gamma \models_v (\forall x)\Diamond P(x).$$

On the other hand, suppose we had

$$\mathcal{M}, \Gamma \models_v \Diamond(\forall x)P(x).$$

Then we would have one of

$$\mathcal{M}, \Delta \models_v (\forall x)P(x) \quad \text{or} \quad \mathcal{M}, \Omega \models_v (\forall x)P(x).$$

Say we had the first—the argument for the second is similar. Let w' be the x -variant of v given above— $w'(x) = b$. By item 8 of Definition 8.6.7 again, we must have

$$\mathcal{M}, \Delta \models_{w'} P(x)$$

but we do not, since b is not in $\mathcal{I}(P, \Delta)$.

We have shown that

$$\mathcal{M}, \Gamma \Vdash_v (\forall x)\Diamond P(x) \quad \text{and} \quad \mathcal{M}, \Gamma \nVdash_v \Diamond(\forall x)P(x)$$

and it follows that

$$\mathcal{M}, \Gamma \nVdash_v (\forall x)\Diamond P(x) \supset \Diamond(\forall x)P(x).$$

Hence the sentence is not valid in the model $\mathcal{M}$.

Example 8.6.11 This time we show the converse implication, that is, $\Diamond(\forall x)P(x) \supset (\forall x)\Diamond P(x)$, is valid in all constant domain first-order modal models. Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be any model, let Γ be an arbitrary member of $\mathcal{G}$, and let v be an arbitrary valuation in $\mathcal{M}$. To show

$$\mathcal{M}, \Gamma \Vdash_v \Diamond(\forall x)P(x) \supset (\forall x)\Diamond P(x)$$

it is enough to show that

$$\mathcal{M}, \Gamma \Vdash_v \Diamond(\forall x)P(x) \implies \mathcal{M}, \Gamma \Vdash_v (\forall x)\Diamond P(x).$$

Suppose

$$\mathcal{M}, \Gamma \Vdash_v \Diamond(\forall x)P(x).$$

Then for some Δ with $\Gamma \mathcal{R} \Delta$,

$$\mathcal{M}, \Delta \Vdash_v (\forall x)P(x),$$

and so if w is any x -variant of v in $\mathcal{M}$,

$$\mathcal{M}, \Delta \Vdash_w P(x),$$

but $\Gamma \mathcal{R} \Delta$, and hence

$$\mathcal{M}, \Gamma \Vdash_w \Diamond P(x).$$

Since w was *any* x -variant of v , we have

$$\mathcal{M}, \Gamma \Vdash_v (\forall x)\Diamond P(x).$$

Exercises

Exercise 8.6.1 Assume all models are **K** models, that is, no special conditions are placed on the accessibility relation. Which of the following sentences are valid in all constant domain first-order modal **K** models and which are not?

1. $[(\exists x)\Diamond P(x) \wedge \Box(\forall x)(P(x) \supset Q(x))]\supset (\exists x)\Diamond Q(x)$.
2. $(\forall x)\Box P(x) \supset \Box(\forall x)P(x)$.
3. $\Box(\forall x)P(x) \supset (\forall x)\Box P(x)$.
4. $(\exists x)\Box P(x) \supset \Box(\exists x)P(x)$.
5. $\Box(\exists x)P(x) \supset (\exists x)\Box P(x)$.
6. $(\exists x)\Diamond[\Box P(x) \supset (\forall x)\Box P(x)]$.
7. $(\exists x)\Diamond[P(x) \supset (\forall x)\Box P(x)]$.
8. $(\exists x)(\forall y)\Box R(x, y) \supset (\forall y)\Box(\exists x)R(x, y)$.

Exercise 8.6.2 Give a proof of Proposition 8.6.8. You should use complete induction on the degree of Φ .

8.7 Varying Domain Models

There is a more general notion of model than the constant domain version. One can allow quantifier domains to vary from world to world. This gives us *actualist quantification*. Think of the domain associated with a world as what actually exists there, so quantifiers at each world range over the actually existent objects. Allowing domains to vary complicates the machinery somewhat, but not terribly much. Essentially we just replace the domain *set* $\mathcal{D}$ with a domain *function* that can assign a different domain to each world.

Definition 8.7.1 (Varying Domain Skeleton) A structure $\langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ is a *varying domain skeleton* if $\langle \mathcal{G}, \mathcal{R} \rangle$ is a frame and $\mathcal{D}$ is a function mapping members of $\mathcal{G}$ to non-empty sets. The function $\mathcal{D}$ is called a *domain function*. $\langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ is an **L** skeleton if $\langle \mathcal{G}, \mathcal{R} \rangle$ is an **L** frame.

As we said above, think of a domain function as associating with each possible world of the frame the set of things that exist at that world, that is, the set over which quantifiers quantify *at that world*. We often refer to $\mathcal{D}(\Gamma)$ as the domain of the world Γ .

We can think of a constant domain model as a special kind of varying domain model: we simply have a domain function $\mathcal{D}$ that assigns the same set to each possible world. Consequently, anything established about all members of the family of varying domain models will apply automatically to constant domain models. From now on, when appropriate, we will speak indifferently about a constant domain *set* $\mathcal{D}$, or a domain *function* that is constant, with $\mathcal{D}$ as its value at each world. The difference won't matter.

We do face a significant complication before we can properly define varying domain models though. Suppose a formula $\Box(P(x) \vee \neg P(x))$ is true at some possible world Γ , under a valuation v that assigns c to x , where c is something in the domain of world Γ . If $\Box$ is to have its standard interpretation, then for any world Δ that is accessible from Γ , we should have that $P(x) \vee \neg P(x)$ is true at Δ under the valuation v . But, *how do we know that $v(x)$, that is, c , exists at Δ ?* More precisely, how do we know $c \in \mathcal{D}(\Delta)$? The answer, of course, is that we don't. And so we find ourselves required to consider the truth of a formula at a world when free variables of the formula have values that may not exist at the world in question!

There are two ways out of this problem. One is to allow *partial* models: take $P(x) \vee \neg P(x)$ to be neither true nor false at Δ under valuation v , when $v(x)$ is not in the domain associated with Δ . The other approach, which is the one we follow, is to say that even though $v(x)$ might not exist in the domain associated with Δ , it does exist under alternative circumstances we are willing to consider, and consequently talk about $v(x)$ is meaningful at Δ . Then at Δ , either the property P is true of $v(x)$ or is false of it, and in any event, $P(x) \vee \neg P(x)$ holds. As we said, this is the approach we follow.

Definition 8.7.2 (Frame Domain) Let $\mathcal{F} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ be a varying domain skeleton. The *domain of the frame* is the set $\cup\{\mathcal{D}(\Gamma) \mid \Gamma \in \mathcal{G}\}$. That is, we put together the domains associated with all the possible worlds of the frame. We write $\mathcal{D}(\mathcal{F})$ for the domain of the frame $\mathcal{F}$.

In a varying domain frame $\mathcal{F} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$, if $\Gamma \in \mathcal{G}$ we can think of $\mathcal{D}(\Gamma)$ as the set of things that actually exist at world Γ , or in situation Γ , or in state Γ . And we can think of $\mathcal{D}(\mathcal{F})$ as the things it makes sense to talk about at Γ , though these things may or may not exist at Γ .

Note that if our varying domain frame happens to be constant domain, if $\mathcal{D}$ assigns the same domain to each possible world, what we are now calling the domain of the frame turns out to be the domain of the frame as we used the terminology in Sect. 8.6.

The definition of interpretation now is essentially what it was before (Definition 8.6.2), with obvious minor modifications. It will later be extended starting with Definition 14.2.1.

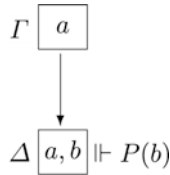
Definition 8.7.3 (Interpretation) $\mathcal{I}$ is an *interpretation* in varying domain skeleton $\mathcal{F} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ if $\mathcal{I}$ assigns, to each n -place relation symbol R , and to each possible world $\Gamma \in \mathcal{G}$, some n -place relation on the domain $\mathcal{D}(\mathcal{F})$ of the frame.

Finally, the definition of model is word for word as it was earlier (Definition 8.6.3), except that we use varying domain skeletons instead of constant domain ones.

Definition 8.7.4 (Model) A varying domain *first-order model* is a structure $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ where $\langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ is a varying domain skeleton and $\mathcal{I}$ is an interpretation in it. We say $\mathcal{M}$ is a varying domain first-order model for a modal logic $\mathbf{L}$ if the frame $\langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ is an $\mathbf{L}$ skeleton.

Incidentally, just as with constant domain models, we use the terminology *domain of a model* and mean by it the domain of the underlying frame. We write $\mathcal{D}(\mathcal{M})$ for the domain of the model $\mathcal{M}$.

Example 8.7.5 Here is an example of a varying domain first-order model. Let $\mathcal{G}$ consist of two possible worlds, Γ and Δ , with $\Gamma \mathcal{R} \Delta$, and $\mathcal{R}$ holding in no other cases. Let $\mathcal{D}(\Gamma) = \{a\}$, and $\mathcal{D}(\Delta) = \{a, b\}$. Let P be a one-place relation symbol. Finally, let $\mathcal{I}(P, \Gamma)$ be the empty set (that is, nothing is in this relation), and let $\mathcal{I}(P, \Delta)$ consist of just b . This specifies a varying domain first-order model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$. We represent it schematically as follows.



We continue exploring this example in Example 8.7.9.

To define truth in varying domain models we modify earlier machinery slightly, to ensure that quantifiers really do quantify over things that exist.

Definition 8.7.6 (Valuation) As before, a *valuation* in a varying domain model $\mathcal{M}$ is a mapping v that assigns to each free variable x some member $v(x)$ of the domain of the model $\mathcal{D}(\mathcal{M})$.

The following is the varying domain version of Definition 8.6.6.

Definition 8.7.7 (Variant) Let v and w be two valuations. We say a valuation w is an x -variant of v at Γ if v and w agree on all variables except possibly the variable x , and also $w(x)$ is a member of $\mathcal{D}(\Gamma)$.

The following will be further extended later, starting with Definition 14.2.3.

Definition 8.7.8 (Truth in a Varying Domain Model) Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a varying domain first-order modal model. For each $\Gamma \in \mathcal{G}$ and each valuation v in $\mathcal{D}(\mathcal{M})$:

1–7 Exactly as in Definition 8.6.7.

8. $\mathcal{M}, \Gamma \Vdash_v (\forall x)\Phi \iff$ for every x -variant w of v at Γ , $\mathcal{M}, \Gamma \Vdash_w \Phi$.

9. $\mathcal{M}, \Gamma \Vdash_v (\exists x)\Phi \iff$ for some x -variant w of v at Γ , $\mathcal{M}, \Gamma \Vdash_w \Phi$.

The essential difference between the definition above and the earlier Definition 8.6.7 is that here we require x -variants to assign to x something that is in the domain associated with the possible world we are considering. This, of course, was unnecessary for constant domain models, since all worlds had the same domains.

As before, the behavior of a formula at a world with respect to a valuation is not affected by the values assigned to variables that are not free in the formula. More precisely, Proposition 8.6.8 carries over to the varying domain case. Consequently

whether a *sentence* is true or false at a world is completely independent of which valuation we may choose, and so we can suppress mention of valuations when dealing with sentences.

Example 8.7.9 We continue with Example 8.7.5, and show sentence $\Diamond(\exists x)P(x) \supset (\exists x)\Diamond P(x)$ is not valid in the model we gave. This is what is called a Barcan formula, something that will be discussed further in Sect. 8.10.

Let v be any valuation in $\mathcal{M}$. Let w be like v on all variables except that $w(x) = b$. Now, $\mathcal{I}(P, \Delta)$ is the one-place relation that holds of just b , and $w(x) = b$, so by definition, we have

$$\mathcal{M}, \Delta \Vdash_w P(x).$$

Since $b \in \mathcal{D}(\Delta)$, w is an x -variant of v at Δ and it follows that we have

$$\mathcal{M}, \Delta \Vdash_v (\exists x)P(x)$$

and consequently we have

$$\mathcal{M}, \Gamma \Vdash_v \Diamond(\exists x)P(x).$$

On the other hand, suppose we had

$$\mathcal{M}, \Gamma \Vdash_v (\exists x)\Diamond P(x).$$

Then for some x -variant w of v at Γ we would have

$$\mathcal{M}, \Gamma \Vdash_w \Diamond P(x).$$

But $w(x)$ must be a , since a is the only member of the domain associated with Γ . It follows that

$$\mathcal{M}, \Delta \Vdash_w P(x)$$

which can only happen if $w(x)$ is in the relation $\mathcal{I}(P, \Delta)$, but $w(x) = a$, and a is not in this relation. This contradiction shows our supposition was wrong, and so

$$\mathcal{M}, \Gamma \not\Vdash_v (\exists x)\Diamond P(x).$$

Now it follows that

$$\mathcal{M}, \Gamma \not\Vdash_v \Diamond(\exists x)P(x) \supset (\exists x)\Diamond P(x)$$

and so the sentence $\Diamond(\exists x)P(x) \supset (\exists x)\Diamond P(x)$ is not true at Γ , and hence is not valid in the model $\mathcal{M}$.

Exercises

Exercise 8.7.1 Which of the sentences of Exercise 8.6.1 are valid in all *varying domain* models and which are not?

8.8 Free Logic, Briefly

Varying domain semantics is often referred to as *actualist*. The domain associated with a possible world can be thought of as what actually exists at that world, while the domain of the model represents what does exist or might exist had circumstances been different. At each world quantifiers range over the domain of that world so quantification is over what actually exists, while free variables can take on any value in the domain of the model, that is, values that might exist. This is quite reasonable because it is natural to talk about things that might exist but don't, we just don't include them among existents. There is a still more general structure that appears in the literature. After all, we sometimes talk about things that can't exist, perhaps because their properties are contradictory. We wouldn't want to put these in the domain of any possible world, so we never include them as existents under any circumstances, but we might still want to allow them as values assignable to free variables.

There is something called *free logic* that was originally introduced by Karel Lambert in the 1960s in a classical setting, and it can easily be adapted to a modal one. We do not follow the free logic route here, but we mention it for your general information. For classical logic, part of the free logic idea is for models to have two domains, an *inner* one and an *outer* one, where the first is a subset of the second. Quantifiers range over the inner domain, which might be empty, while free variables can be assigned values from the outer domain. Effectively, our varying domain models are like free logics at each world, with the inner domain being that of the possible world and the outer domain being the domain of the model. But we could have allowed an outer domain that is larger than the domain of the model. And something similar could also have been done for constant domain semantics.

We do not follow the route just sketched above. Basically we don't because it introduces complexities that we don't need for us to make the points we wish to make here. But you should know that things can be done differently, and free logic based modal models are available if needed for some purpose. One should think of the invention of semantic machinery as the creation of tools to help us understand the behavior of philosophical concepts. The machinery is something we create, and not something out there waiting for us to discover. It is a useful design guide to have a semantics that is as simple as can be managed, provided it is sufficient for the particular purpose intended.

Here are some references to free logic, for further reading: Nolt (2021), Bencivenga (1986), Bencivenga (2002), Lambert (2003), Lambert (2001), and Reicher (2016).

8.9 Different Media, Same Message

We have now seen two radically different semantics: constant domain and varying domain. Which is, or should be, primary? It turns out, in a precise but technical sense, that it doesn't matter. We can formalize the same philosophical ideas either way, with a certain amount of care.

In varying domain semantics we think of the domain of a world as what actually exists at that world, in that state, under those circumstances. A quantifier ranges over what actually exists. Thus $(\exists x) \dots$ means, *there is something, x , that actually exists such that* $\dots$. The domain of the *model* consists of what exists at all worlds, collectively, and so represents what *might* exist. While quantifiers do not range over the domain of the model, free variables take their values there. Thus a formula $\Phi(x)$, with x free, generally has a different behavior from its universal closure, $(\forall x)\Phi(x)$, in marked contrast to classical logic.

Constant domain semantics, on the other hand, has the same domain for every world. This common domain is analogous to the domain of the model when varying domain semantics is used. Then in the constant domain approach, quantifiers range over what does and what might exist. Now $(\exists x) \dots$ can be read, “there is something, x , that could exist, such that $\dots$.”

Suppose, in constant domain semantics, we had an “existence predicate” that tells us which of the things that could exist actually do exist and which do not. Then we could relativize constant domain quantifiers to this predicate, and achieve an effect rather like being in a varying domain model. Based on the development in Hughes and Cresswell (1996), this can be made precise as follows.

Definition 8.9.1 (Existence Relativization) Let $\mathcal{E}$ be a one-place relation symbol. (We intend this symbol to be an existence primitive, and will not use it for any other purpose.) The *existence relativization* of a formula Φ , denoted $\Phi^{\mathcal{E}}$, is defined by the following conditions.

1. If Φ is atomic, $\Phi^{\mathcal{E}} = \Phi$.
2. $(\neg\Phi)^{\mathcal{E}} = \neg(\Phi^{\mathcal{E}})$.
3. For a binary connective $\circ$, $(\Phi \circ \Psi)^{\mathcal{E}} = (\Phi^{\mathcal{E}} \circ \Psi^{\mathcal{E}})$.
4. $(\Box\Phi)^{\mathcal{E}} = \Box\Phi^{\mathcal{E}}$.
5. $(\Diamond\Phi)^{\mathcal{E}} = \Diamond\Phi^{\mathcal{E}}$.
6. $((\forall x)\Phi)^{\mathcal{E}} = (\forall x)(\mathcal{E}(x) \supset \Phi^{\mathcal{E}})$;
7. $((\exists x)\Phi)^{\mathcal{E}} = (\exists x)(\mathcal{E}(x) \wedge \Phi^{\mathcal{E}})$.

The intuition is that the relativized quantifiers are restricted by a predicate intended to mean: actually exists. Now, the following says to what extent this stratagem succeeds.

Proposition 8.9.2 *Let Φ be a sentence not containing the symbol $\mathcal{E}$. Then: Φ is valid in every varying domain model if and only if $\Phi^{\mathcal{E}}$ is valid in every constant domain model for which the extension of the predicate $\mathcal{E}$ is non-empty at each world.*

Proof The proof of this Proposition amounts to formally applying the informal motivating ideas, and doing so in a straightforward way.

Part I. Suppose Φ is not valid in some varying domain model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$. We construct a constant domain model in which $\mathcal{E}$ meets the non-emptiness condition and in which $\Phi^{\mathcal{E}}$ is not valid. This will establish half the Proposition.

Let $\mathcal{M}'$ be the constant domain model $\langle \mathcal{G}', \mathcal{R}', \mathcal{D}', \mathcal{I}' \rangle$ constructed as follows. $\mathcal{G}'$ and $\mathcal{R}'$ are the same as $\mathcal{G}$ and $\mathcal{R}$ respectively. $\mathcal{D}'$ is the domain of the original model $\mathcal{M}$. $\mathcal{I}'$ is like $\mathcal{I}$ on all relation symbols except $\mathcal{E}$, and for that, $\mathcal{I}'(\mathcal{E}, \Gamma) = \mathcal{D}(\Gamma)$ (which is required to be non-empty). Note that since the varying domain model $\mathcal{M}$ and the constant domain model $\mathcal{M}'$ have the same model domains, any valuation in one model is a valuation in both.

Now to complete Part I it is enough to show the following. For any formula Φ not containing $\mathcal{E}$ (possibly with free variables, though), for any world $\Gamma \in \mathcal{G}$, and for any valuation v ,

$$\mathcal{M}, \Gamma \Vdash_v \Phi \iff \mathcal{M}', \Gamma \Vdash_v \Phi^{\mathcal{E}}.$$

This is proved by induction on the complexity of Φ . The base case, where Φ is atomic, is directly by the construction of $\mathcal{M}'$ —and in particular the definition of $\mathcal{I}'$. The various induction cases are straightforward. We give the existential quantifier case only. Thus, suppose Φ is $(\exists x)\Psi$, and the result is known for Ψ .

Suppose first that $\mathcal{M}, \Gamma \Vdash_v (\exists x)\Psi$. Then for some x -variant w of v at Γ , $\mathcal{M}, \Gamma \Vdash_w \Psi$, so by the induction hypothesis we have $\mathcal{M}', \Gamma \Vdash_w \Psi^{\mathcal{E}}$. Since w is an x -variant at Γ , $w(x) \in \mathcal{D}(\Gamma)$. Then by definition, $w(x) \in \mathcal{I}'(\mathcal{E}, \Gamma)$, so $\mathcal{M}', \Gamma \Vdash_w \mathcal{E}(x)$. Thus we have $\mathcal{M}', \Gamma \Vdash_w \mathcal{E}(x) \wedge \Psi^{\mathcal{E}}$, so $\mathcal{M}', \Gamma \Vdash_v (\exists x)(\mathcal{E}(x) \wedge \Psi^{\mathcal{E}})$.

Conversely, if we assume $\mathcal{M}', \Gamma \Vdash_v (\exists x)(\mathcal{E}(x) \wedge \Psi^{\mathcal{E}})$, it follows by a similar argument that $\mathcal{M}, \Gamma \Vdash_v (\exists x)\Psi$, and this completes the existential quantifier case for Part I.

Part II. Suppose $\Phi^{\mathcal{E}}$ is not valid in some constant domain model in which $\mathcal{E}$ meets the non-emptiness condition. A varying domain model in which Φ is not valid must be constructed. We leave this to you as an exercise.

Essentially, Proposition 8.9.2 says that instead of working with varying domain models we could work with constant domain models, provided we suitably relativize quantifiers. Relativization provides us an embedding of varying domain semantics into constant domain semantics. We will see below and in Chap. 12 that there is a way of going in the other direction as well, though it is more complicated.

Since each of varying domain and constant domain semantics can simulate the other, semantical machinery does not dictate a solution to us for the problem of what quantifiers must quantify over. Whether we take quantifiers as ranging over the actually existent or over the possibly existent is, in a precise sense, just a manner of speaking. Which way to speak is, finally, a choice to be made based on what seems most natural for what one wants to say.

In this work we treat both varying and constant domain systems—both actualist and possibilist quantification. For some of what we do, it won't matter which version we choose. When the formal details differ we will say so, and specify which version is appropriate at that point.

Exercises

Exercise 8.9.1 Let Φ be the formula

$$(\exists x)(\Box P(x) \supset \Box(\forall x)P(x))$$

where $P(x)$ is atomic. What is $\Phi^{\mathcal{E}}$?

Exercise 8.9.2 Give the argument for Part II of Proposition 8.9.2.

8.10 Barcan and Converse Barcan Formulas

We have seen that there is a natural embedding of varying domain modal logic into the constant domain version, using quantifier relativization to an existence predicate $\mathcal{E}$. Going the other way is more complex, but the route goes through interesting territory. Further, it is territory whose exploration began early in the development of quantified modal logic, and whose connections with the issues of present concern to us were not realized until some time later.

The modal operators $\Box$ and $\Diamond$ are like disguised quantifiers. To say Φ is necessary at a world is to say Φ is true at *all* accessible worlds; to say Φ is possible is to say Φ is true at *some* accessible world. The move to first-order modal logic adds the quantifiers $\forall$ and $\exists$. We know that in classical first-order logic some quantifier permutations are allowed, but others are not: $(\forall x)(\forall y)\Phi$ and $(\forall y)(\forall x)\Phi$ are equivalent; $(\exists y)(\forall x)\Phi$ implies $(\forall x)(\exists y)\Phi$; $(\forall x)(\exists y)\Phi$ does not imply $(\exists y)(\forall x)\Phi$. Things like these are important facts. A natural question, then, is: what permutations hold between the conventional first-order quantifiers and the modal operators?

Marcus (1946) is the first study of quantified modal logic. Since first-order modal semantics had not yet been invented, all work at the time was axiomatic. It turned out that a particular assumption concerning quantifier/modality permutability was found to be useful. That assumption, or rather, a modernized version of it, has come

to be called the *Barcan formula*. (Barcan was the name Marcus was known by at the time of writing the paper Marcus (1946).) Properly speaking, it is not a formula but a scheme.

Definition 8.10.1 (Barcan Formula) All formulas of the following forms are Barcan formulas:

1. $(\forall x)\Box\Phi \supset \Box(\forall x)\Phi$;
2. $\Diamond(\exists x)\Phi \supset (\exists x)\Diamond\Phi$.

There is some (convenient) redundancy in the definition above.

$$\Diamond(\exists x)\Phi \supset (\exists x)\Diamond\Phi.$$

is a Barcan formula of form 2. It is equivalent to its contrapositive:

$$\neg(\exists x)\Diamond\Phi \supset \neg\Diamond(\exists x)\Phi.$$

But $\neg\Diamond\psi \equiv \Box\neg\psi$, and $\neg(\exists x)\psi \equiv (\forall x)\neg\psi$, so this formula in turn is equivalent to

$$(\forall x)\Box\neg\Phi \supset \Box(\forall x)\neg\Phi,$$

and this is a Barcan formula of form 1. In fact, if we take *either* form as basic, we can derive the other. We simply adopt them both, for convenience.

Over the years it has become common to refer to implications that go the other way as *Converse Barcan* formulas. It was observed that, for certain natural ways of axiomatizing first-order modal logics, Converse Barcan formulas were provable, though this was not the case for Barcan formulas. Consequently Converse Barcan formulas were thought to be of lesser importance than Barcan formulas, since only assuming the truth of Barcan formulas seemed to make a difference. Eventually this point of view proved misleading, and both versions are now seen to play significant roles.

Definition 8.10.2 (Converse Barcan Formula) All formulas of the following forms are Converse Barcan formulas:

1. $\Box(\forall x)\Phi \supset (\forall x)\Box\Phi$;
2. $(\exists x)\Diamond\Phi \supset \Diamond(\exists x)\Phi$.

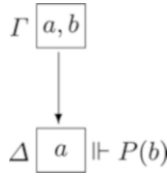
There is the same redundancy in our definition of Converse Barcan formula that there was for Barcan formulas themselves.

The situation concerning Barcan and Converse Barcan formulas is clarified by possible world semantics. It turns out that these formulas really correspond to fundamental semantic properties of frames, and so the fact that they turned up from time to time over the years was no coincidence. We are about to present the semantical connections, but first a convenient piece of terminology. If we say the Barcan formula is valid (or valid under certain circumstances), we mean *all* Barcan

formulas are; if we say the Barcan formula is not valid, we mean *at least one* Barcan formula is not. Similarly for the Converse Barcan formula. It is customary to speak of these formulas as if they were single entities.

The first thing to observe is that neither the Barcan formula nor the Converse Barcan formula is valid generally, that is, in the family of all varying domain **K** models. We have already seen this for the Barcan formula, in Example 8.7.5, continued in Example 8.7.9. For the Converse Barcan formula, Example 8.10.3 below gives a model in which $(\exists x)\Diamond P(x) \supset \Diamond(\exists x)P(x)$ fails. We leave it to you to verify this.

Example 8.10.3 A varying domain counterexample to the Converse Barcan Formula $(\exists x)\Diamond P(x) \supset \Diamond(\exists x)P(x)$.



Now we introduce two classes of frames intermediate between constant domain and varying domain. They turn up frequently in the literature and are very natural mathematically. Philosophically, the monotonic condition, Definition 8.10.4, is quite plausible in some situations. Temporally, once something has come into existence, one can make a good case that it can be talked about forever afterward. The antimonotonic condition, Definition 8.10.7, is less natural philosophically, but certainly it has as good a technical interest as monotonicity does. We will examine some of the technical issues as we proceed.

Definition 8.10.4 (Monotonicity) A varying domain skeleton $\langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ is *monotonic* provided, for every $\Gamma, \Delta \in \mathcal{G}$, if $\Gamma \mathcal{R} \Delta$ then $\mathcal{D}(\Gamma) \subseteq \mathcal{D}(\Delta)$. A model is monotonic if its frame is.

Example 8.7.5, continued in Example 8.7.9, is varying domain and, as it happens, monotonic. It shows that $\Diamond(\exists x)P(x) \supset (\exists x)\Diamond P(x)$ is not valid in all monotonic models. However, its converse is.

Example 8.10.5 We show $(\exists x)\Diamond\Phi(x) \supset \Diamond(\exists x)\Phi(x)$ is valid in all varying domain models that are monotonic. Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a monotonic model. We show

$$\mathcal{M}, \Gamma \Vdash_v (\exists x)\Diamond\Phi(x) \implies \mathcal{M}, \Gamma \Vdash_v \Diamond(\exists x)\Phi(x)$$

where $\Gamma \in \mathcal{G}$ and v is a valuation in $\mathcal{D}(\mathcal{M})$.

Assume

$$\mathcal{M}, \Gamma \Vdash_v (\exists x)\Diamond\Phi(x),$$

then for some x -variant w of v at Γ ,

$$\mathcal{M}, \Gamma \Vdash_w \Diamond \Phi(x)$$

and hence for some Δ such that $\Gamma \mathcal{R} \Delta$,

$$\mathcal{M}, \Delta \Vdash_w \Phi(x).$$

Now, $\Gamma \mathcal{R} \Delta$ and the model is monotonic, so $\mathcal{D}(\Gamma) \subseteq \mathcal{D}(\Delta)$. The valuation w is an x -variant of v at Γ , so $w(x) \in \mathcal{D}(\Gamma)$, and hence $w(x) \in \mathcal{D}(\Delta)$. Then w is also an x -variant of v at Δ , so we have

$$\mathcal{M}, \Delta \Vdash_v (\exists x)\Phi(x)$$

and hence

$$\mathcal{M}, \Gamma \Vdash_v \Diamond(\exists x)\Phi(x).$$

It turns out this Example is no coincidence. There is an exact correspondence between the Converse Barcan formula and monotonicity.

Proposition 8.10.6 *A varying domain skeleton is monotonic if and only if every model based on it is one in which the Converse Barcan formula is valid.*

Proof One direction has already been done. If the skeleton is monotonic the Converse Barcan formula must be valid in every model based on it, as was shown in Example 8.10.5 (in that example, $\Phi(x)$ is really any formula).

For the other direction, suppose $\langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ is *not* monotonic. We produce a model based on it in which one particular Converse Barcan formula is not valid.

Since $\langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ is not monotonic, there must be some $\Gamma, \Delta \in \mathcal{G}$ with $\Gamma \mathcal{R} \Delta$, but $\mathcal{D}(\Gamma) \not\subseteq \mathcal{D}(\Delta)$. Say $c \in \mathcal{D}(\Gamma)$ but $c \notin \mathcal{D}(\Delta)$. Let P be a one-place relation symbol, and define an interpretation $\mathcal{I}$ as follows. $c \in \mathcal{I}(P, \Delta)$, but for any $\Omega \in \mathcal{G}$ other than Δ , $\mathcal{I}(P, \Omega)$ is empty. We claim that, in the model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$,

$$\mathcal{M}, \Gamma \not\Vdash (\exists x)\Diamond P(x) \supset \Diamond(\exists x)P(x).$$

Let v be any valuation in $\mathcal{M}$, and let w be the x -variant of it such that $w(x) = c$. Since $c \in \mathcal{I}(P, \Delta)$,

$$\mathcal{M}, \Delta \Vdash_w P(x);$$

since $\Gamma \mathcal{R} \Delta$,

$$\mathcal{M}, \Gamma \Vdash_w \Diamond P(x);$$

and since $c \in \mathcal{D}(\Gamma)$, w is an x -variant of v at Γ , so

$$\mathcal{M}, \Gamma \Vdash_v (\exists x) \Diamond P(x).$$

On the other hand, $(\exists x)P(x)$ is not true at *any* world. More precisely, if Ω is any member of $\mathcal{G}$,

$$\mathcal{M}, \Omega \not\Vdash_v (\exists x)P(x).$$

For otherwise, for some x -variant v' of v at Ω , we would have

$$\mathcal{M}, \Omega \Vdash_{v'} P(x)$$

and so $v'(x) \in \mathcal{I}(P, \Omega)$. This is not the case if $\Omega \neq \Delta$ because then $\mathcal{I}(P, \Omega)$ is empty. And this is not the case if $\Omega = \Delta$ because the only member of $\mathcal{I}(P, \Delta)$ is c , so $v'(x)$ would have to be c , but $c \notin \mathcal{D}(\Delta)$, so v' would not be an x -variant of v at Δ .

Now, since $(\exists x)P(x)$ is not true at any world of the model, it is not true at any world accessible from Γ , and hence

$$\mathcal{M}, \Gamma \not\Vdash_v \Diamond(\exists x)P(x)$$

and this establishes our claim.

Notice that while monotonicity gets us the Converse Barcan formula, it is not enough to ensure the Barcan formula itself, as Examples 8.7.5 and 8.7.9 show. We now turn to something that does.

Definition 8.10.7 (Anti-Monotonicity) A varying domain skeleton $\langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ is *anti-monotonic* provided, for $\Gamma, \Delta \in \mathcal{G}$, $\Gamma \mathcal{R} \Delta$ implies $\mathcal{D}(\Delta) \subseteq \mathcal{D}(\Gamma)$. (Note that the order of inclusion is reversed.)

Proposition 8.10.8 *A varying domain skeleton is anti-monotonic if and only if every model based on it is one in which the Barcan formula is valid.*

We leave it to you to prove this, as Exercise 8.10.2.

We still have not brought constant domain frames into the picture. The problem is, in a constant domain skeleton, $\mathcal{D}(\Gamma) = \mathcal{D}(\Delta)$ for all Γ, Δ , whether or not $\Gamma \mathcal{R} \Delta$. But it can happen that a frame consists of two parts that are not connected. That is, the accessibility relation does not hold between any possible world in one of the parts and any possible world in the other. Then the truth or falsity of a formula throughout one of the parts can have no bearing on what happens at a possible world in the other part. There is no formula whose truth in such a disconnected model could force the possible worlds in one of the disconnected parts to have the same domains as possible worlds in the other. However there is a weaker notion than constant domain that is natural and does give us something we can use here.

Definition 8.10.9 (Locally Constant Domain) Call a varying domain skeleton $\langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ *locally constant domain* provided, for $\Gamma, \Delta \in \mathcal{G}$, if $\Gamma \mathcal{R} \Delta$ then $\mathcal{D}(\Gamma) = \mathcal{D}(\Delta)$. A model is locally constant domain if its frame is.

It follows from the combination of Propositions 8.10.6 and 8.10.8 that having *both* the Barcan formula and the Converse Barcan formula valid corresponds to the *locally constant domain* condition. The following puts the final piece in place.

Proposition 8.10.10 *A sentence Φ is valid in all locally constant domain models if and only if Φ is valid in all constant domain models.*

Proof One direction is trivial. If Φ is valid in all locally constant domain models, among these are all the true constant domain models, so it is valid in all of them as well.

In the other direction, suppose Φ is a sentence that is *not* valid in some varying domain model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ that meets the locally constant domain condition; say Φ is not true at $\Gamma \in \mathcal{G}$. We will produce a constant domain model $\mathcal{M}' = \langle \mathcal{G}', \mathcal{R}', \mathcal{D}', \mathcal{I}' \rangle$ in which Φ is also not valid.

Let us say there is a *path* in the model $\mathcal{M}$ from Δ_1 to Δ_2 if there is a sequence of worlds, starting with Δ_1 , finishing with Δ_2 , with each world in the sequence accessible from the one before it. And let us say a world Δ_2 is *relevant* to Δ_1 if Δ_2 is Δ_1 itself, or if there is a path from Δ_1 to Δ_2 . (The case of Δ_2 being Δ_1 can be treated as a special case of the ‘path’ clause: we simply have a path of length 0.)

Now, let $\mathcal{G}'$ be the subset of $\mathcal{G}$ consisting of those possible worlds relevant to Γ . Let the accessibility relation $\mathcal{R}'$ be the relation $\mathcal{R}$ restricted to $\mathcal{G}'$, $\mathcal{D}'$ be domain function $\mathcal{D}$ restricted to $\mathcal{G}'$, and interpretation $\mathcal{I}'$ be $\mathcal{I}$ restricted to $\mathcal{G}'$. We thus have a submodel $\mathcal{G}'$ of $\mathcal{G}$. (In the literature this is commonly called a *generated* submodel.) Since there is a path from Γ to every member of $\mathcal{G}'$, it follows easily from the fact that $\mathcal{M}$ meets the locally constant domain condition, that $\mathcal{D}'$ assigns the same domain to every member of $\mathcal{G}'$, and thus $\mathcal{G}'$ is, in fact, constant domain.

We now show the following, by induction on formula complexity. For each formula Ψ , for each valuation v in $\mathcal{M}'$ (which will also be a valuation in $\mathcal{M}$), and for each $\Delta \in \mathcal{G}'$ (which will also be a member of $\mathcal{G}$):

$$\mathcal{M}', \Delta \Vdash_v \Psi \iff \mathcal{M}, \Delta \Vdash_v \Psi. \quad (\dagger)$$

Condition $(\dagger)$ is immediate for Ψ atomic, by definition of $\mathcal{G}'$. The propositional cases are straightforward, and are left to the reader. Now suppose Ψ is $(\forall x)\Pi(x)$, where $(\dagger)$ is known for simpler formulas. We show the right-left implication of $(\dagger)$, leaving the easier left-right implication to the reader. Assume $\mathcal{M}, \Delta \Vdash_v (\forall x)\Pi(x)$ where v is a valuation in $\mathcal{G}'$, and hence maps variables to the constant domain of model $\mathcal{M}'$. Then $\mathcal{M}, \Delta \Vdash_w \Pi(x)$ where w is an arbitrary x -variant of v such that $w(x) \in \mathcal{D}(\Delta)$. Since $\Delta \in \mathcal{G}'$, $\mathcal{D}'(\Delta) = \mathcal{D}(\Delta)$ and hence $w(x)$ is in the constant domain of $\mathcal{M}'$. Then by the induction hypothesis for $(\dagger)$, $\mathcal{M}', \Delta \Vdash_w \Pi(x)$. Since w was an arbitrary x -variant, it follows that $\mathcal{M}, \Delta \Vdash_v (\forall x)\Pi(x)$. This concludes (our part of) the induction.

Finally, suppose Φ is any sentence that is not valid in varying domain $\mathcal{M}$. Say Φ fails at possible world Γ , that is, $\mathcal{M}, \Gamma \not\models_v \Phi$, for some valuation v . Since Φ is a sentence it has no free variables, and so the choice of v does not matter. Let v be any valuation that is actually in the submodel $\mathcal{M}'$. Construct the constant domain submodel $\mathcal{M}'$, as above. Then by $(\dagger)$, Φ will also fail in the constant domain model $\mathcal{M}'$ at Γ .

Remarkably, as we have seen, simple conditions on frames correspond exactly to the Barcan and the Converse Barcan formulas. Thus these formulas have an importance that goes considerably beyond the technical issue of quantifier permutation. They really say something about the existence assumptions we are making in our semantics. The Converse Barcan formula says that, as we move to an alternative situation, nothing passes out of existence. The Barcan formula says that, under the same circumstances, nothing comes into existence. The two together say the same things exist no matter what the situation.

Prior (1957) raised doubts about the Barcan formula. On the temporal reading it says that if everything that *now* exists will at all future times be Φ , then at all future times everything that *then* exists will be Φ . But this holds generally only if no new things come into existence, and things are always coming into existence. It is important to see that Prior's quantifier is actually relativized to *things that now exist* and *things that then exist*. As we saw in Sect. 8.9, relativising the possibilist quantifier effectively turns it into the actualist quantifier. Where there is no relativization, as when we take the quantifier to range over all objects that have been, are, or will be, the Barcan formula will hold.

The very same distinction is operating in the following seeming counterexample to the claim that the Barcan formula holds in every constant domain model. The example is from epistemic logic. Even if

$$\text{Everything is known to be } F \quad (8.24)$$

is true, nonetheless, it does not follow that

$$\text{It is known that everything is } F \quad (8.25)$$

is also true. The reason is, we might not know that we have everything. Analogously, we might be able to prove of each number that it is F without being able to prove that every number is F . In fact the quantifier is tacitly relativized in (8.24) to what is known to exist, and what is known to exist need not be the same as what does exist (known or not) in a world compatible with every fact that is known.

In Sect. 8.9 we saw how to embed varying domain modal logic into the constant domain version. We now have the possibility of going the other way. We briefly sketch how this can be done. We do not give formal details, since it is not an issue that is fundamental here.

We introduced a somewhat complicated notion of logical consequence for propositional modal logic in Sect. 5.6. That notion extends naturally to a first-

order version. Recall, there were really two kinds of deduction assumption: local and global. Based on what was established above, it is not hard to establish that a sentence Φ is valid in all constant domain models if and only if Φ is a consequence of the Barcan and the Converse Barcan formulas as *global* assumptions, in varying domain logic. This allows us to turn questions of validity for constant domain logic into corresponding questions about varying domain logic. Unfortunately, things are not as simple as we would like. Both the Barcan and the Converse Barcan formulas are really *schemes*, with infinitely many instances. For a given sentence Φ , which instances should we try working with? Fortunately, we will see that once equality has been introduced, both the families of Barcan and Converse Barcan formulas collapse to *single* instances (Sect. 12.8). This fact, finally, will give us useful embedding machinery from constant to varying domain logic.

Exercises

Exercise 8.10.1 Which of the sentences of Exercise 8.6.1 are valid in all *monotonic domain* models and which are not? Similarly for *anti-monotonic domain* models.

Exercise 8.10.2 Prove Proposition 8.10.8.

Exercise 8.10.3 Show the Converse Barcan formula need not be valid in a model whose frame is anti-monotonic.

Exercise 8.10.4 Here are four formula schemes:

1. $(\exists x)\Box\Phi(x) \supset \Box(\exists x)\Phi(x)$
2. $\Box(\exists x)\Phi(x) \supset (\exists x)\Box\Phi(x)$
3. $(\forall x)\Diamond\Phi(x) \supset \Diamond(\forall x)\Phi(x)$
4. $\Diamond(\forall x)\Phi(x) \supset (\forall x)\Diamond\Phi(x)$

Just as we gave two versions of the Barcan formula, and observed they came in pairs, the same is true for the schemes above. Determine which pairs of schemes constitute equivalent assumptions.

Exercise 8.10.5 For the formula schemes in Exercise 8.10.4, determine the status of their validity assuming: constant domains; varying domains; monotonic domains; anti-monotonic domains.

References

- Bencivenga, E. (1986). *Free logics* (first publication). (chap. 6, vol. 3, pp. 373–426). In D. M. Gabbay & F. Guentner, 1983–1989. Cambridge: Kluwer.
- Bencivenga, E. (2002). *Free logics* (second publication). (chap. 3, vol. 5, pp. 147–196). In D. M. Gabbay & F. Guentner, 2001 to present. Berlin: Springer.

- Gabbay, D. M., & Guentner, F. (Eds.). (1983–1989). *Handbook of philosophical logic*. Four volumes. Dordrecht: Kluwer.
- Gabbay, D. M., & Guentner, F. (Eds.). (2001 to present). *Handbook of philosophical logic* (2nd edn.). Springer. Multiple volumes. Dordrecht: Kluwer.
- Hughes, G. E., & Cresswell, M. J. (1996). *A new introduction to modal logic*. London: Routledge.
- Kripke, S. (1963). Semantical considerations on modal logic. *Acta Philosophica Fennica*, 16, 83–94.
- Lambert, K. (2001). *The Blackwell guide to philosophical logic*. In L. Goble (Ed.), (Chap. Free Logics). Oxford: Blackwell.
- Lambert, K. (2003). *Free logic: Selected essays*. Cambridge: Cambridge University Press.
- Lewis, D. (1968). Counterpart theory and quantified modal logic. *The Journal of Philosophy*, 65, 113–126.
- Marcus, R. B. (1946). A functional calculus of first order based on strict implication. *Journal of Symbolic Logic*, 11, 1–16.
- Nolt, J. (2021). Free logic. In E. N. Zalta (Ed.), *The Stanford encyclopedia of philosophy* (Fall 2021). Stanford: Metaphysics Research Lab, Stanford University.
- Prior, A. N. (1957). *Time and modality*. Oxford: Clarendon Press.
- Quine, W. V. O. (1943). Notes on existence and necessity. *The Journal of Philosophy*, 40, 113–127.
- Quine, W. V. O. (1953). Three grades of modal involvement. In *The ways of paradox and other essays* (pp. 156–174). New York: Random House.
- Quine, W. V. O. (1961). *From a logical point of view* (2nd. rev.). New York: Harper & Row.
- Reicher, M. (2016). Nonexistent objects. In E. N. Zalta (Ed.), *The Stanford encyclopedia of philosophy*. Stanford: Metaphysics Research Lab, Stanford University.
- Smullyan, R. M., & Fitting, M. (1996). *Set theory and the continuum problem* (rev. edn.), Dover Publications, 2010. Errata at <http://melvinfitting.org/errata/errata.html>. Oxford: Oxford University Press.
- Whitehead, A. N. & Russell, B. (1925–1927). *Principia mathematica* (2nd edn.). Three volumes. Cambridge: Cambridge University Press.

Chapter 9

First-Order Modal Tableaux



In Sects. 7.1 and 7.2 we gave prefixed tableau rules for propositional modal logics in what we called the Lesser Modal Cube. Now we extend these to deal with quantifiers. But we have considered two versions of quantifier semantics: varying domain and constant domain. Not surprisingly, these correspond to different versions of tableau rules for quantifiers. Also not surprisingly, the rules corresponding to constant domain semantics are simpler, so we will start with them.

The basic idea behind all quantifier rules is a natural one, and is well-known for classical logic—see the canonical Smullyan (1968) or Fitting (1996) or Priest (2008) for more extensive treatments. If, in some classical model $(\forall x)P(x)$ is true, then $P(x)$ is true no matter what x stands for in the domain of the model so, speaking informally, we should be able to replace x with anything. On the other hand, if $(\exists x)P(x)$ is true, some value of x in the domain of the model makes $P(x)$ true. Following standard mathematical procedure, we can introduce a *name* for an object making x true, say p , and conclude that $P(p)$ is true. But of course, p should be a name that has no prior commitments—we should not have used it for anything previously. We guarantee this “newness” by two devices. First, we create a completely fresh list of free variables, called parameters, just for the purpose of instantiating quantifiers during tableau proofs. Since they are new, they can not have already appeared in the formula and so cannot have been quantified. Thus an instantiation using a parameter is automatically substitutable, Definition 8.1.6. Second, whenever we introduce a parameter into a tableau proof to instantiate an existential quantifier, we require that it has not appeared previously in that particular proof (or at least, in that branch of the proof that concerns us).

Adapting this treatment of quantifiers to modal logic is rather straightforward. The only complications come from whether we want a single list of parameters (appropriate for constant domain semantics), or multiple lists (appropriate for varying domain semantics). There are no surprises in the details.

9.1 Constant Domain Modal Tableaus

In constant domain models, quantifiers range over the same domain no matter what possible world we consider. For tableaus, this means we want a *single* list of additional free variables.

Definition 9.1.1 (Constant Domain Parameters) In addition to the list of free variables (see Sect. 8.1), we now add a second list of *new* free variables. These are called *parameters*. They may appear in formulas in the same way as the original list of free variables *but we will never quantify them*. We use $p, q, r, \dots$ to represent arbitrary parameters.

One of the conditions in the definition above is that, while parameters are free variables, they are never quantified. As we noted above, they are then always substitutable for any standard variable in any context where it is free, see Definition 8.1.6.

As usual, a tableau proof of a formula X is a closed tableau starting with $\neg X$, where closure means that each branch contains some formula and its negation. We place an important restriction on things: *we only construct tableaus that start with formulas having no free variables*. That is, we provide tableau proofs only for closed formulas, also called *sentences*. There are ways around this, but they complicate things and we avoid the issue to keep things relatively simple. Note that since parameters are not allowed to be bound, they cannot appear in the formulas we are trying to prove, since these cannot contain free variables. Parameters can, however, be introduced in the course of a tableau construction.

Propositional rules are exactly as before, in Sects. 7.1 and 7.2. The particular choice of a modal logic doesn't usually matter for what we say here. We leave it open and don't mention it, unless it plays a significant role.

In the following quantifier rules we use our notational convention that $\Phi(x)$ is a formula with some (possibly no) occurrences of the free variable x , and $\Phi(p)$ is the result of replacing *all free occurrences* of x with occurrences of the parameter p . Now, here are the rules that are to be added to our earlier propositional ones to produce a tableau system for *constant domain* modal logics. (Of course which modal logic depends on which propositional rules are used.)

Definition 9.1.2 (Universal Rules—Constant Domain) In the following, p is any parameter whatsoever.

$$\frac{\sigma (\forall x)\Phi(x)}{\sigma \Phi(p)} \qquad \frac{\sigma \neg(\exists x)\Phi(x)}{\sigma \neg\Phi(p)}$$

Definition 9.1.3 (Existential Rules—Constant Domain) In the following, p is a parameter that is *new to the tableau branch*.

$$\frac{\sigma (\exists x)\Phi(x)}{\sigma \Phi(p)} \qquad \frac{\sigma \neg(\forall x)\Phi(x)}{\sigma \neg\Phi(p)}$$

Notice that since we start proofs with sentences, Definition 8.1.7, which contain no free variables, and only parameters can be introduced during the course of a proof, the only free variables that can appear throughout a tableau proof are parameters. This actually plays a role in the soundness argument, and it is very helpful to have terminology for it.

Definition 9.1.4 We call a formula *pseudo-closed* if the only free variables it contains are parameters.

Then, all formulas that appear in tableau proofs that involve quantification are pseudo-closed.

Example 9.1.5 Here is a proof using the constant domain tableau rules, of a Barcan formula. We use the propositional rules for **K**.

- 1 $\neg[\Diamond(\exists x)\Phi(x) \supset (\exists x)\Diamond\Phi(x)]$ 1.
- 1 $\Diamond(\exists x)\Phi(x)$ 2.
- 1 $\neg(\exists x)\Diamond\Phi(x)$ 3.
- 1.1 $(\exists x)\Phi(x)$ 4.
- 1.1 $\Phi(p)$ 5.
- 1 $\neg\Diamond\Phi(p)$ 6.
- 1.1 $\neg\Phi(p)$ 7.

In this, 2 and 3 are from 1 by a Conjunctive Rule; 4 is from 2 by a Possibility Rule; 5 is from 4 by an Existential Rule (note, p is a parameter, and it is new at the point of introduction in line 5); 6 is from 3 by a Universal Rule (p is not new now, but for a Universal Rule anything is allowed); and 7 is from 6 by a Necessity Rule.

Example 9.1.6 This time we show a constant domain non-proof, of

$$[\Diamond(\exists x)P(x) \wedge \Diamond(\exists x)Q(x)] \supset (\exists x)\Diamond[P(x) \wedge Q(x)].$$

Once again the propositional rules are those of **K**. The proof is displayed in Fig. 9.1.

Here 2 and 3 are from 1 by a Conjunctive Rule, as are 4 and 5 from 2; 6 is from 4 and 7 is from 5 by a Possibility Rule; 8 is from 6 and 9 is from 7 by the Existential Rule (notice, both parameters are new at the point of introduction); 10 and 11 are from 3 by the Universal Rule; 12 and 13 are from 10, and 14 and 15 are from 11 by the Necessity Rule; 16 and 17 are from 12 by a Disjunctive Rule; as are 18 and 19 from 15; 20 and 21 from 13; 22 and 23 from 14; and 24 and 25 also from 14.

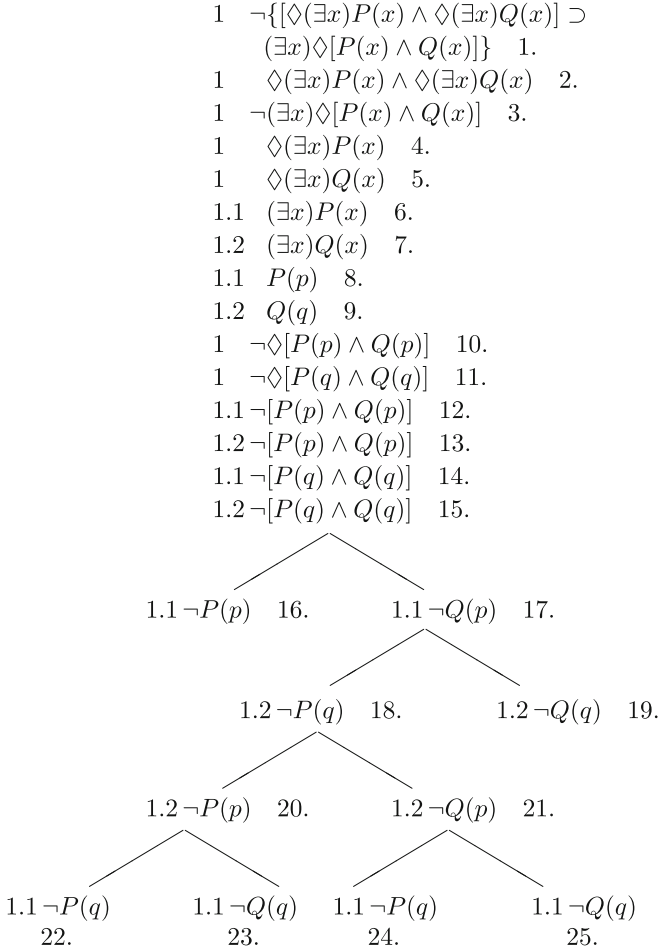


Fig. 9.1 Example of a non-proof

The branches ending with 16 and 19 are closed, but the other four branches are not. Note that this by itself does not establish the sentence is unprovable, since it is not the only tableau we could have constructed.

Exercises

Exercise 9.1.1 Show the interdefinability of quantifiers can be proven using tableaux. That is, prove the following (use the **K** rules):

1. $(\forall x)\Phi(x) \supset \neg(\exists x)\neg\Phi(x)$;
2. $\neg(\exists x)\neg\Phi(x) \supset (\forall x)\Phi(x)$;

3. $(\exists x)\Phi(x) \supset \neg(\forall x)\neg\Phi(x)$;
4. $\neg(\forall x)\neg\Phi(x) \supset (\exists x)\Phi(x)$.

Exercise 9.1.2 In Exercise 8.6.1 you were asked to determine **K** validity of various sentences. Now determine which of them have tableau proofs using the propositional **K** rules and the constant domain quantifier rules.

9.2 Varying Domain Tableaus

For tableaus corresponding to varying domain semantics there is one important change that is needed: we introduce a whole *family* of lists of parameters, one for each prefix, rather than a single list. More specifically, we assume that to each prefix σ there is associated an infinite list of free variables, still called parameters, in such a way that distinct prefixes never have the same parameter associated with them.

We will write p_σ to indicate that p is a parameter associated with the prefix σ . For two such parameters p_σ and p_τ to be the same, both p and q must be the same, and σ and τ must be the same. Now we give the varying domain quantifier rules.

Definition 9.2.1 (Universal Rules—Varying Domain) In the following, p_σ can be any parameter that is associated with the prefix σ .

$$\frac{\sigma (\forall x)\Phi(x)}{\sigma \Phi(p_\sigma)} \qquad \frac{\sigma \neg(\exists x)\Phi(x)}{\sigma \neg\Phi(p_\sigma)}$$

Definition 9.2.2 (Existential Rules—Varying Domain) In the following, p_σ is a parameter associated with the prefix σ , subject to the condition that p_σ is new to the tableau branch.

$$\frac{\sigma (\exists x)\Phi(x)}{\sigma \Phi(p_\sigma)} \qquad \frac{\sigma \neg(\forall x)\Phi(x)}{\sigma \neg\Phi(p_\sigma)}$$

Example 9.2.3 We give a varying domain **K** tableau proof of

$$[\Diamond(\exists x)\Box A(x) \wedge \Box(\forall x)\Diamond B(x)] \supset \Diamond(\exists x)\Diamond[A(x) \wedge B(x)].$$

$$\begin{array}{lcl}
1 & \neg\{[\Diamond(\exists x)\Box A(x) \wedge \Box(\forall x)\Diamond B(x)] \supset \\
& \quad \Diamond(\exists x)\Diamond[A(x) \wedge B(x)]\} & 1. \\
1 & \Diamond(\exists x)\Box A(x) \wedge \Box(\forall x)\Diamond B(x) & 2. \\
1 & \neg\Diamond(\exists x)\Diamond[A(x) \wedge B(x)] & 3. \\
1 & \Diamond(\exists x)\Box A(x) & 4. \\
1 & \Box(\forall x)\Diamond B(x) & 5. \\
1.1 & (\exists x)\Box A(x) & 6. \\
1.1 & (\forall x)\Diamond B(x) & 7. \\
1.1 & \neg(\exists x)\Diamond[A(x) \wedge B(x)] & 8. \\
1.1 & \Box A(p_{1.1}) & 9. \\
1.1 & \Diamond B(p_{1.1}) & 10. \\
1.1 & \neg\Diamond[A(p_{1.1}) \wedge B(p_{1.1})] & 11. \\
1.1.1 & B(p_{1.1}) & 12. \\
1.1.1 & A(p_{1.1}) & 13. \\
1.1.1 & \neg[A(p_{1.1}) \wedge B(p_{1.1})] & 14. \\
& \swarrow \quad \searrow & \\
1.1.1 & \neg A(p_{1.1}) & 15. \\
1.1.1 & \neg B(p_{1.1}) & 16.
\end{array}$$

Here 2 and 3 are from 1 by a Conjunctive Rule; as are 4 and 5 from 2; 6 is from 4 by a Possibility Rule; 7 is from 5 and 8 from 3 by a Necessity Rule; 9 is from 6 by an Existential Rule; 10 is from 7 and 11 is from 8 by a Universal Rule; 12 is from 10 by a Possibility Rule; 13 is from 9 and 14 is from 11 by a Necessity Rule; and finally 15 and 16 are from 14 by a Disjunctive Rule.

And we also give an example of a non-proof—of a Barcan formula.

Example 9.2.4 The following is a failed varying domain **K** attempt to prove the Barcan formula: $(\forall x)\Box A(x) \supset \Box(\forall x)A(x)$.

$$\begin{array}{lcl}
1 & \neg[(\forall x)\Box A(x) \supset \Box(\forall x)A(x)] & 1. \\
1 & (\forall x)\Box A(x) & 2. \\
1 & \neg\Box(\forall x)A(x) & 3. \\
1.1 & \neg(\forall x)A(x) & 4. \\
1.1 & \neg A(p_{1.1}) & 5. \\
1 & \Box A(q_1) & 6. \\
1.1 & A(q_1) & 7. \\
1 & \Box A(r_1) & 8. \\
1.1 & A(r_1) & 9. \\
& \vdots &
\end{array}$$

In this, 2 and 3 are from 1 by a Conjunctive Rule; 4 is from 3 by a Possibility Rule; 5 is from 4 by an Existential Rule; 6 is from 2 by a Universal Rule (notice that we can *not* use 1.1 as a subscript, since the prefix involved is 1); 7 is from 6 by a Necessity Rule; 8 is from 2 by a Universal Rule; 9 is from 8 by a Necessity Rule; and so on. We can run through all the parameters with a subscript of 1, but there will be no closure.

We will return to this again, in Examples 9.4.2 and 9.6.1.

Exercises

Exercise 9.2.1 Give a **K** varying domain tableau proof of the following.

$$[\Box(\forall x)\Diamond(\exists y)P(x, y) \wedge \Diamond(\exists x)\Box(\forall y)Q(x, y)] \supset \Diamond(\exists x)\Diamond(\exists y)[P(x, y) \wedge Q(x, y)]$$

Exercise 9.2.2 Attempt to prove a Converse Barcan formula using the varying domain tableau rules. Explain why you cannot.

Exercise 9.2.3 Starting with the varying domain tableau rules:

1. Design tableau **K** rules appropriate for proving the formulas valid in all monotonic varying domain models, Definition 8.10.4;
2. Do the same for anti-monotonic domain models, Definition 8.10.7.

9.3 Varying Domain Tableau Soundness

Of course we want our tableau rules to exactly capture the quantifier behavior specified by first-order modal models. That is, we must prove *soundness* and *completeness* of the various tableau systems allowing quantifiers. Doing this continues the work we did in the propositional case. In this section we show soundness and *we present proofs only for the varying domain tableau rules*. The constant domain rules are similar, simpler, and we omit the details.

Recall that soundness means: anything provable is valid. Although tableaux only prove *sentences*, formulas that contain parameters turn up in proofs so we must take them into account. Earlier we said that the notion of a prefix function, Definition 7.4.1, would grow more elaborate as tableau machinery was added. That is now happening.

Definition 9.3.1 (Prefix Function, Extended) Let S be a set of prefixed pseudo-closed modal formulas, Definition 9.1.4, and let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a varying domain modal model, with v being a valuation in it. A *prefix function* for S in $\mathcal{M}$,

using v , is a mapping, θ , assigning to those prefixes that occur in S possible worlds in $\mathcal{G}$ so that,

1. If σ and $\sigma.n$ both occur as prefixes in S then $\theta(\sigma.n)$ is a world that is accessible from $\theta(\sigma)$, that is, $\theta(\sigma)\mathcal{R}\theta(\sigma.n)$.
2. If the parameter p_σ occurs in S then $v(p_\sigma) \in \mathcal{D}(\theta(\sigma))$.

Now the following extends Definition 7.4.2.

Definition 9.3.2 (Satisfiable) Let S be a set of pseudo-closed prefixed formulas (Definition 9.1.4). S is *satisfiable* in the varying domain model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$, with respect to valuation v , if there is a *prefix function* using v , call it θ , such that: whenever $\sigma \Phi$ is in S , then $\mathcal{M}, \theta(\sigma) \Vdash_v \Phi$.

The rest of Definition 7.4.2 is carried over directly. A tableau branch is satisfiable if the set of prefixed formulas on it is satisfiable (in some model, with respect to some valuation); a tableau is satisfiable if some branch of it is satisfiable.

Propositions 3.3.2 said classically and 7.4.3 said modally that a propositional tableau that was closed could not be satisfiable. This carries over to the first-order case directly, and with the same proof.

Next we need a version of Propositions 7.4.4 and 3.3.3 that takes quantifiers into account. The wording is exactly the same, and the proof builds on that of the earlier propositions. **L** is any of the modal logics in the Lesser Modal Cube.

Proposition 9.3.3 *If an **L** tableau branch extension rule (including varying domain quantifier rules) is applied to a satisfiable **L** tableau, the result is another **L** satisfiable tableau.*

Proof The proof has several cases, depending on what branch extension rule was applied. All the propositional and modal cases are treated exactly as in the proof of Proposition 7.4.4, and don't need to be repeated. The new thing is the quantifier cases, and there are only two of these.

First the universal case. Suppose we have a satisfiable tableau branch, $\sigma (\forall x)\Phi(x)$ is on it, and we add $\sigma \Phi(p_\sigma)$ to the end. Say the original branch was satisfiable in model $\mathcal{M}$ at possible world $\theta(\sigma)$ using the valuation v . Then in particular, under this valuation $(\forall x)\Phi(x)$ is true at $\theta(\sigma)$. According to Definition 8.7.8, it must be that $\Phi(x)$ is also true under any x -variant of v , provided it assigns to x a value in the domain of $\theta(\sigma)$. But v itself will assign a value to p_σ that is in the domain of $\theta(\sigma)$, so the lengthened branch is still satisfiable at the same world, using the same valuation that we originally used.

Second, the more complicated existential case. Again suppose we have a satisfiable tableau branch, $\sigma (\exists x)\Phi(x)$ is on it, and we add $\sigma \Phi(p_\sigma)$ to the end, where p_σ is new to the branch. And again, say the original branch was satisfiable in model $\mathcal{M}$ using the valuation v . This time v itself won't work for what we need. It assigns some member of the domain of $\theta(\sigma)$ to p_σ , but there is no reason why it would be a member for which $\Phi(x)$ is true at $\theta(\sigma)$. We need to make some changes.

Since $\sigma (\exists x)\Phi(x)$ was on the branch before it was extended, and that branch is satisfiable using valuation v , there is some x variant w of v that assigns to x a value

in the domain of $\theta(\sigma)$ and is such that $\Phi(x)$ is true using w . Loosely, $w(x)$ is a value for which $\Phi(x)$ is true (using v to assign values to variables other than x). Suppose we change valuation v to valuation v' by redefining it at p_σ to be whatever $w(x)$ is, which is known to be a thing making $\Phi(x)$ true at $\theta(\sigma)$. Since v' agrees with v on all variables except p_σ , and p_σ was new and hence didn't occur in any formula on the original branch, everything from the original branch is still satisfiable using valuation v' . But now $\Phi(p_\sigma)$ will also be true using v' . That is, the extended branch is satisfiable, but using v' instead of v .

With Proposition 9.3.3 established, tableau soundness follows immediately, using precisely the same argument we gave in the propositional case—see Theorem 7.4.5.

9.4 Hintikka Sets (Again)

We are heading toward a proof of the completeness of first-order modal tableaux. For propositional modal tableaux and for classical tableaux, Hintikka sets were a useful tool, and that is still the case. Once again we present the details only for the varying domain version since the constant domain version is similar and simpler.

Classical propositional Hintikka sets were characterized in Definition 3.4.1, and modal versions for logics in the Lesser Modal Cube in Definition 7.5.1. Our quantified modal version builds on this background; we, therefore, do not repeat it. But note, when the clauses from these earlier Definitions are brought in, it is with the understanding that the language now is the first-order one specified in Sect. 8.1.

Definition 9.4.1 (First Order Varying Domain Hintikka Set) Let $\mathbf{L}$ be one of the logics in the Lesser Modal Cube. A set H is a *varying domain $\mathbf{L}$ Hintikka set* provided the members of H are prefixed pseudo-closed formulas, Definition 9.1.4, H meets the conditions of Definition 7.5.1, and in addition the following applies. (The numbering continues that of Definition 7.5.1.)

5. H meets the following quantifier closure conditions:

$$\begin{aligned} \sigma (\exists x)\Psi(x) \in H &\implies \sigma \Psi(p_\sigma) \in H \text{ for some parameter } p_\sigma, \\ \sigma \neg(\forall x)\Psi(x) \in H &\implies \sigma \neg\Psi(p_\sigma) \in H \text{ for some parameter } p_\sigma, \\ \sigma (\forall x)\Psi(x) \in H &\implies \sigma \Psi(p_\sigma) \in H \text{ for every parameter } p_\sigma, \\ \sigma (\neg\exists x)\Psi(x) \in H &\implies \sigma \neg\Psi(p_\sigma) \in H \text{ for every parameter } p_\sigma. \end{aligned}$$

In condition 5 the clause for the universal quantifier (and also for the negated existential quantifier) uses every parameter with a σ subscript. This is in the interests of simplicity of presentation. In fact, it would be enough to just use those parameters that appear in H . (If there are none, pick an arbitrary parameter and use it.)

Sometimes this can allow us to create a finite counter-model, but it is a side issue for us here.

Example 9.4.2 Here is an example of a **K** Hintikka set. It is, in fact, the set of prefixed formulas on the open tableau branch shown in Example 9.2.4.

$$\{1 \neg[(\forall x)\Box A(x) \supset \Box(\forall x)A(x)], 1 (\forall x)\Box A(x), 1 \neg\Box(\forall x)A(x), \\ 1.1 \neg(\forall x)A(x), 1.1 \neg A(p_{1.1}), 1 \Box A(q_1), 1.1 A(q_1), 1 \Box A(r_1), 1.1 A(r_1), \dots\}$$

We will continue this in Example 9.6.1.

Proposition 9.4.3 (Hintikka's Lemma) *Let L be a varying domain first order modal logic whose modal operators meet the L conditions of the Lesser Modal Cube. Let H be a first order varying domain L Hintikka set. Then H is satisfiable in a varying domain L model.*

Proof The proof begins like that of Proposition 7.5.2, then extra parts are added to take care of quantifiers. We present things assuming that L is **K**; the other cases are similar but with straightforward complications. We use Hintikka set H to construct a model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$, and then show H is satisfiable in it.

Exactly as in the propositional version, we let the set of possible worlds $\mathcal{G}$ be the set of prefixes that occur in H . And if σ and $\sigma.n$ are both in $\mathcal{G}$, set $\sigma \mathcal{R} \sigma.n$. (This part needs modification for logics other than **K**. See the Proof of Proposition 7.5.2.) Now we come to the new material.

For each $\sigma \in \mathcal{G}$, take $\mathcal{D}(\sigma)$ to be the set of all parameters p_σ having subscript σ (an infinite set). And finally for each k -place relation symbol P , define an interpretation function as follows.

$$\mathcal{I}(P, \sigma) = \{\langle t_1, \dots, t_k \rangle \mid \sigma P(t_1, \dots, t_k) \text{ occurs in } H\}$$

We have now defined a model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$.

Define a valuation v_0 by setting $v_0(p_\sigma) = p_\sigma$ for each parameter p_σ and letting v_0 be arbitrary on free variables that are not parameters. Also define a prefix mapping θ by $\theta(\sigma) = \sigma$ for each $\sigma \in \mathcal{G}$. Note that then each prefix also functions as a possible world of our model and the definition of θ says each prefix designates itself, so in what follows we generally omit explicit mention of θ . We claim that each prefixed formula in H is satisfied in $\mathcal{M}$ with respect to the valuation v_0 , using the mapping θ . More precisely we show the following, which is similar to (★★) from the proof of Proposition 7.5.2 (Propositional Hintikka's Lemma), except that it allows for quantifiers in formulas.

$$\begin{aligned} &\text{if } \sigma \Phi \in H \text{ then } \mathcal{M}, \sigma \Vdash_{v_0} \Phi \\ &\text{if } \sigma \neg\Phi \in H \text{ then } \mathcal{M}, \sigma \not\Vdash_{v_0} \Phi \end{aligned} \tag{★★★}$$

This is shown by Complete Induction on the complexity of the formula Φ . The quantifier cases and the cases having to do with atomic formulas (which now have structure) are all that we discuss here, since the other cases are exactly as in the proof of Proposition 7.5.2.

The base case is atomic formulas, and their negations. Suppose $\sigma P(t_1, \dots, t_k)$ is in H , where P is a k -place relation symbol. By definition, $\mathcal{J}(P, \sigma)$ contains $\langle t_1, \dots, t_k \rangle$. Since each t_i must be a parameter, and $v_0(t_i) = t_i$, it follows immediately that $\mathcal{M}, \sigma \Vdash_{v_0} P(t_1, \dots, t_k)$.

For negations of atomic formulas, if $\sigma \neg P(t_1, \dots, t_k)$ is in H then $\sigma P(t_1, \dots, t_k)$ is not, by condition 1 for Hintikka Sets back in Definition 7.5.1. So $\langle t_1, \dots, t_k \rangle$ is not in $\mathcal{J}(P, \sigma)$, and it follows that $\mathcal{M}, \sigma \nVdash_{v_0} P(t_1, \dots, t_k)$.

Now for the induction cases. Suppose Φ is not atomic, and $(\star \star \star)$ is known for simpler formulas, that is, formulas of lower degree—we show it for Φ itself. As we said, the cases covering the propositional connectives and the modal operators are the same as in earlier chapters. We only consider the quantifier cases in detail.

Suppose $\sigma (\forall x)\Psi(x)$ is in H , and $(\star \star \star)$ is known for simpler formulas. By Definition 9.4.1, $\sigma \Psi(p_\sigma)$ is in H for every parameter p_σ that is in $\mathcal{D}(\sigma)$. Since $\Psi(p_\sigma)$ is of lower degree than $(\forall x)\Psi(x)$ the induction hypothesis applies, and we have that $\mathcal{M}, \sigma \Vdash_{v_0} \Psi(p_\sigma)$ holds for every parameter p_σ in $\mathcal{D}(\sigma)$. Now, let w be any x -variant of v_0 at σ . By definition of $\mathcal{D}(\sigma)$, $w(x)$ must be some parameter of the form p_σ . Since $\mathcal{M}, \sigma \Vdash_{v_0} \Psi(p_\sigma)$, it follows that $\mathcal{M}, \sigma \Vdash_w \Psi(x)$. Finally, since w was an arbitrary x -variant of v_0 at σ , by definition, $\mathcal{M}, \sigma \Vdash_{v_0} (\forall x)\Psi(x)$.

The negated universal quantifier case is similar, as are the existential cases, and we leave them to you.

Exercises

Exercise 9.4.1 Construct a Hintikka set that contains *both* $\neg[(\forall x)\Box A(x) \supset \Box(\forall x)A(x)]$ and $\neg[(\Box(\forall x)A(x) \supset (\forall x)\Box A(x))]$.

9.5 Tableau Completeness with Quantification

For both propositional classical and modal tableaux, one of the ways we showed completeness was by extracting a model from a failed tableau proof. Of course it must be a tableau in which we have done enough to give us what we need to specify a model. This involves the construction of a tableau in which every rule that could be applied, at some point was.

For propositional classical tableaux things were quite simple. Assuming we never applied the same rule to a formula more than once on a branch, all tableau proof attempts were finite. It was clear when we had done everything that could be

done. Once modal operators were added this still applied for many modal logics, but there were some exceptions such as **S4**, for which proof attempts could run forever. There are ways of deciding when we can stop a construction, but they add complexity to the systematic tableau construction algorithm, and unless we have some particular reason for producing finite possible world models, we can simply ignore the issue. With quantifiers added too, it is very commonly the case that failed proof attempts will be infinite. It is possible to fine tune a Systematic Tableau Construction Algorithm so that it generates finite Hintikka sets when possible, or at least much of the time when possible. The resulting algorithm descriptions are more complex, and are not necessary since all we are interested in here is producing a counter-model to an unprovable formula. We don't particularly care if there is a finite one. For other purposes this may be important, but here, in order to have a more easily describable algorithm, we simply ignore any niceties about generating finite Hintikka sets and hence finite counter-models. For us, any counter-models we produce via a Systematic Algorithm will simply have the set of all parameters as the domain of the model.

We now present a proof search algorithm for varying domain logics. If $\sigma (\forall x)\Phi(x)$ occurs on a branch and the branch never closes, our procedure will run through all the infinitely many parameters. A constant domain version is similar but simpler, and we leave you to think about it. Also we present the details for **K**. Other modal logics can be handled similarly but we also leave that to you.

Systematic Tableau Construction Algorithm for K There is an infinite list of parameters associated with each prefix, and there are infinitely many prefixes possible. But we have a *countable* alphabet for our logical language, and this implies that it is possible to combine all parameters, no matter what prefix is involved as a subscript, into a single list: $\rho_1, \rho_2, \rho_3, \dots$ (The subscripts you see here are not the associated prefixes; they just mark the position of the parameter in the list.) Thus, for each prefix, and for each parameter associated with that prefix, that parameter occurs in the list somewhere. (A proof that this can be done involves some simple set theory, and would take us too far afield here. Take our word for it—this can be done.)

What we present is not the only systematic construction possible, but we just need one. It goes in *stages*. Assume we have a closed modal formula Φ for which we are trying to find a **K** tableau proof. For stage 1 we simply put down $1 \rightarrow \Phi$, getting a one-branch, one-formula tableau.

Having completed stage n , if the tableau construction has not terminated here is what to do for stage $n + 1$. Assume there are b open branches. Number the open branches from left to right, $1, 2, \dots, b$. Process branch 1, then branch 2, and so on. When branch number b has been processed, this completes stage $n + 1$.

To process an open branch, go through it from bottom to top, processing it prefixed formula by prefixed formula. Each step may add new prefixed formulas to the end of the branch, or even split the end to produce two longer branches. Since processing the original branch proceeds from its bottom to its top, any new

formulas that are added to branch ends are not processed at this stage, but are left for the next stage.

To process a prefixed formula occurrence, what to do depends on the form it takes. Let us say we have an occurrence of the prefixed formula Φ . Then for each branch passing through that occurrence of Φ do the following.

1. If Φ is $\sigma \neg \neg \Psi$, add $\sigma \Psi$ to the branch end, unless it is already on the branch.
2. If Φ is $\sigma \Psi \wedge \Omega$, add $\sigma \Psi$ to the branch end unless it is already present on the branch, and similarly for $\sigma \Omega$. The other conjunctive cases are treated in the same way.
3. If Φ is $\sigma \Psi \vee \Omega$, and if neither $\sigma \Psi$ nor $\sigma \Omega$ occurs on the branch, split the end of the branch and add $\sigma \Psi$ to one fork and $\sigma \Omega$ to the other. The other disjunctive cases are treated in the same way.
4. If Φ is $\sigma \Diamond \Psi$, and if $\sigma.k \Psi$ does not occur on the branch for any k , choose the *smallest* integer k such that the prefix $\sigma.k$ does not appear on the branch at all, and add $\sigma.k \Psi$ to the branch end. Similarly for the negated necessity case.
5. If Φ is $\sigma \Box \Psi$, add to the end of the branch every prefixed formula of the form $\sigma.k \Psi$ where this prefixed formula does not already occur on the branch, but $\sigma.k$ does occur as a prefix somewhere on the branch. Similarly for the negated possibility case.
6. If Φ is $\sigma (\exists x) \Psi(x)$, and if $\sigma \Psi(p_\sigma)$ does not occur on the branch for any parameter with σ as a subscript, then choose the *first* parameter ρ_i in the list $\rho_1, \rho_2, \dots$ having σ as a subscript, and add $\sigma \Psi(\rho_i)$ to the branch end. Similarly for the negated universal case.
7. If Φ is $\sigma (\forall x) \Psi(x)$, add to the end of the branch the prefixed formula $\sigma \Psi(\rho_i)$ where: ρ_i is the first parameter in the list $\rho_1, \rho_2, \dots$ having σ as a subscript, but where $\sigma \Psi(\rho_i)$ does not already occur on the branch. Similarly for the negated existential case.

When all open branches have been processed, from left to right, if the tableau is closed then terminate the construction. Likewise if the tableau is not closed, but stage $n + 1$ added no new prefixed formulas, terminate the construction. Otherwise, repeat.

We have now completed the description of our systematic tableau construction procedure. It is complicated, and we do not expect people to really construct tableaus this way since there is no place for intuition or insight. But it does guarantee that everything that could be tried will sooner or later be tried. And this is all we care about now. It is a *fair* algorithm.

There are some important remarks we need to make, before moving on to the completeness proof itself. If this systematic construction procedure were to produce a closed tableau, it would be a proof of Φ . If it does not produce a closed tableau, the procedure could still terminate because it ran out of things to do. (Try constructing a systematic proof for $(\forall x)P(x)$. It will terminate very quickly, but without a proof.) If the construction does terminate, we can always choose an open branch. But it can happen that the construction does not terminate because we do not find a proof, yet

we do not run out of new things to do. This can easily happen. If a universally quantified formula is present (or a negated existential) item 7 of the algorithm can force us to instantiate using new parameter after new parameter, forever. (Try constructing a systematic proof for $(\exists x)P(x)$ and see what happens.) Thus we could wind up constructing an infinite tree. But even so, there will still be an open branch, *but it will be infinite*. This may seem intuitively correct, but there are subtleties involved that we don't go into here. Fortunately it is not hard to prove. Here is the argument, which is really just a special case of what is called König's Lemma.

Lemma 9.5.1 *If the systematic tableau construction never terminates it will generate a tableau with an infinite open branch.*

Proof In a tableau, a prefixed formula $\sigma \Psi$ that is not at a branch end either has a single prefixed formula immediately below it, or the tableau has a split at $\sigma \Psi$, and there are two prefixed formulas immediately below it. It is customary to call the (one or two) prefixed formulas immediately below $\sigma \Psi$ the *children* of $\sigma \Psi$. Children of a prefixed formula, children of children, children of children of children, and so on, are called *descendants*. Now, call an occurrence of a prefixed formula $\sigma \Psi$ in the tableau we are constructing *good* if the systematic procedure produces infinitely many descendants for it.

Our systematic tableau construction procedure is an attempt to prove some sentence Φ . Suppose the procedure never terminates. Then the prefixed formula $1 \neg\Phi$ we begin with must be good, because every other prefixed formula in the tableau is a descendant of it, and there are infinitely many of them since the construction process does not terminate.

Suppose some prefixed formula occurrence $\sigma \Psi$ in the tableau is good. Then the construction procedure yields infinitely many descendants for $\sigma \Psi$. But $\sigma \Psi$ itself can have at most two children. It follows that at least one of the children of $\sigma \Psi$ must itself have infinitely many descendants. Consequently, if $\sigma \Psi$ is good, $\sigma \Psi$ must have a good child.

Here, then, is why we must have an infinite branch, assuming the systematic procedure does not terminate. The initial prefixed formula, $1 \neg\Phi$, is good. The construction procedure must produce a good child for $1 \neg\Phi$, and this in turn must have a good child, and so on. In this way an infinite branch (of good prefixed formula occurrences) is created.

There is one more important point that can be brought out here. The systematic tableau construction is entirely constructive—at each stage it is fully determined what to do next. But selection of an infinite branch is not constructive at all. A good prefixed formula occurrence must have a good child, but if there are two children, we have no way of knowing which must be good! This is because the definition of goodness is in terms of the completed behavior of the construction procedure, and infinitely many descendants can be produced. At any finite stage we don't generally know which branches will terminate and which will continue, so we don't generally know which parts of the tableau will continue to grow and which will not. This point plays no role in the completeness argument, but it does mean we should not

expect to get a way of *deciding* on provability or non-provability of sentences from our systematic tableau construction procedure. In fact, it can be proved that no such way of deciding is possible, by this or any other means.

We now have what we need to give a proof of completeness for modal tableaus involving varying domain quantification.

Theorem 9.5.2 (K First-Order Varying Domain Tableau Completeness) *If the sentence Φ is valid in all varying domain first-order $\mathbf{K}$ models, Φ has a tableau proof using the varying domain $\mathbf{K}$ rules.*

Proof Suppose Φ is a sentence that has no varying domain tableau proof. We intend to show it is not valid, and to do this we extract a countermodel from a failed attempt at proving Φ . However, not just any failed attempt will do—we might have done something stupid and missed finding a proof that was there all along. Instead, we must be sure we have tried everything that could be tried. So we use the Systematic Tableau Construction Algorithm for $\mathbf{K}$ as described above, beginning the tableau construction with $1 \neg\Phi$.

By our initial supposition that Φ is not provable, our systemic proof attempt for Φ will produce a tableau with an open branch, call it $\mathcal{B}$, possibly finite, possibly infinite. We leave it to you to check that the set of prefixed formulas on $\mathcal{B}$ must be a varying domain Hintikka set, Definition 9.4.1. Then Hintikka's Lemma, Proposition 9.4.3, tells us the set is satisfiable (in a varying domain $\mathbf{K}$ model). The tableau construction began with $1 \neg\Phi$, so this is on every branch of the tableau, in particular on the branch $\mathcal{B}$. Since the set of formulas on $\mathcal{B}$ is satisfiable, there is a varying domain $\mathbf{K}$ model and a possible world of it at which the sentence Φ is false. Thus Φ is not varying domain $\mathbf{K}$ -valid.

We have now proved completeness for the varying domain version of the $\mathbf{K}$ quantifier rules. We note that we have actually proved something stronger than was stated, in two respects. First, we have shown that if a sentence is $\mathbf{K}$ valid, it must have a *systematic* tableau proof. And second, we have shown that if a sentence does not have a $\mathbf{K}$ tableau proof, there is a countermodel *in which domains of different worlds are disjoint*.

Completeness for the constant domain version and for other choices of propositional rules is similar. We leave this to you—see Exercises 9.5.1 and 9.5.2.

Exercises

Exercise 9.5.1 In Sect. 9.1 we gave an example of a failed tableau using the constant domain rules, Example 9.1.6, in which we attempted to prove $[\Diamond(\exists x)P(x) \wedge \Diamond(\exists x)Q(x)] \supset (\exists x)\Diamond[P(x) \wedge Q(x)]$. Use this tableau to construct a constant domain countermodel showing the invalidity of the sentence. Use Example 9.6.1 as a guide, but remember, the result is to be *constant domain*. After you have done this, construct a second countermodel, derived from the same unclosed tableau.

Exercise 9.5.2 State and prove a completeness theorem for the constant domain tableau rules.

9.6 The Completeness Proof, an Example

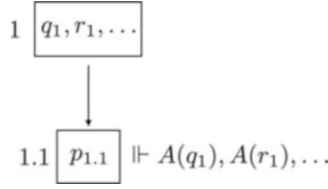
Tableau construction is commonly thought of as a search for a counterexample to a formula. If the search fails, there is no counterexample, hence the formula must be valid. This is what a completeness proof guarantees. Of course the search may not terminate, and so we only “get” a counterexample after infinitely many steps. That is, it may not be possible for a human to tell, at any given point, if an attempted proof search will eventually succeed or not, but if success is unobtainable we can construct a counter-model in principle. We have given the idea motivating tableaux, but it does not give us a decision procedure. Nonetheless, sometimes it is practical to actually carry it out, and in this section we discuss such an example.

Example 9.6.1 In Example 9.2.4 we presented an attempt at a varying domain **K** proof of a Barcan formula, $(\forall x)\Box A(x) \supset \Box(\forall x)A(x)$. For convenience, we repeat the tableau here.

- 1 $\neg[(\forall x)\Box A(x) \supset \Box(\forall x)A(x)]$ 1.
- 1 $(\forall x)\Box A(x)$ 2.
- 1 $\neg\Box(\forall x)A(x)$ 3.
- 1.1 $\neg(\forall x)A(x)$ 4.
- 1.1 $\neg A(p_{1.1})$ 5.
- 1 $\Box A(q_1)$ 6.
- 1.1 $A(q_1)$ 7.
- 1 $\Box A(r_1)$ 8.
- 1.1 $A(r_1)$ 9.
- $\vdots$

There is only one branch, infinite, and not closed. We do not have a proof. The collection of prefixed formulas on the branch is a **K** Hintikka set, see Example 9.4.2. We use it to construct a countermodel to the Barcan formula, following the proof of Proposition 9.4.3. First the possible worlds are, simply, the prefixes that occur in the Hintikka set: $\mathcal{G} = \{1, 1.1\}$. The accessibility relation is the one corresponding to the prefix structure: $1\mathcal{R}1.1$, and $\mathcal{R}$ does not hold otherwise. The domains associated with the possible worlds are just the parameters occurring in the Hintikka set, placed in possible worlds according to subscript: $\mathcal{D}(1) = \{q_1, r_1, \dots\}$ and similarly $\mathcal{D}(1.1) = \{p_{1.1}\}$. Finally, we interpret the one-place relation symbol A the way the Hintikka set says we should. Thus $\mathcal{I}(A, 1.1) = \{q_1, r_1, \dots\}$ because 1.1 $A(q_1)$, 1.1 $A(r_1)$, $\dots$ are in the Hintikka set. And likewise $\mathcal{I}(A, 1) = \emptyset$. We

have created a model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ with two possible worlds. Here is the model displayed schematically.



Now let v_0 be a valuation that assigns, to each parameter, itself: $v_0(r) = r$ for each parameter r (what v_0 does on other free variables won't matter to the argument, and can be left unspecified). And let θ be the prefix mapping that maps each prefix in the Hintikka set to itself: $\theta(1) = 1$ and $\theta(1.1) = 1.1$. We will show the entire Hintikka set is satisfied in $\mathcal{M}$ with respect to the valuation v_0 , using the mapping θ . Once this has been shown, because of line 1 of the tableau we conclude that

$$\mathcal{M}, 1 \Vdash_{v_0} \neg[(\forall x)\Box A(x) \supset \Box(\forall x)A(x)]$$

and hence $(\forall x)\Box A(x) \supset \Box(\forall x)A(x)$ is not a varying domain validity since there is a varying domain model in which it can be falsified.

Here are the details of showing satisfiability for the Hintikka set, which we do making use of the complexity of formulas, from simplest to most complex.

$\mathcal{M}, 1.1 \Vdash_{v_0} A(q_1)$ (Line 7) because $v_0(q_1) = q_1$ and q_1 is in $\mathcal{I}(A, 1.1)$.

$\mathcal{M}, 1.1 \Vdash_{v_0} A(r_1)$ (Line 9) because $v_0(r_1) = r_1$ and r_1 is in $\mathcal{I}(A, 1.1)$.

$\vdots$

$\mathcal{M}, 1 \Vdash_{v_0} \Box A(q_1)$ (Line 6) because 1.1 is the only world accessible from world 1, and we have $\mathcal{M}, 1.1 \Vdash_{v_0} A(q_1)$.

$\mathcal{M}, 1 \Vdash_{v_0} \Box A(r_1)$ (Line 8) because 1.1 is the only world accessible from world 1, and we have $\mathcal{M}, 1.1 \Vdash_{v_0} A(r_1)$.

$\vdots$

$\mathcal{M}, 1.1 \Vdash_{v_0} \neg A(p_{1.1})$ (Line 5) since $v_0(p_{1.1}) = p_{1.1}$ and $p_{1.1}$ is not in $\mathcal{I}(A, 1.1)$, so that $\mathcal{M}, 1.1 \not\Vdash_{v_0} A(p_{1.1})$.

$\mathcal{M}, 1.1 \Vdash_{v_0} \neg(\forall x)A(x)$ (Line 4) since otherwise we would have $\mathcal{M}, 1.1 \Vdash_{v_0} (\forall x)A(x)$. And then if we let w be the x -variant of v_0 such that $w(x) = p_{1.1}$, we would have $\mathcal{M}, 1.1 \Vdash_w A(x)$. But since $v_0(p_{1.1}) = p_{1.1} = w(x)$ we would also have $\mathcal{M}, 1.1 \Vdash_{v_0} A(p_{1.1})$ and we do not.

$\mathcal{M}, 1 \Vdash_{v_0} \neg\Box(\forall x)A(x)$ (Line 3) Because otherwise we would have $\mathcal{M}, 1 \Vdash_{v_0} \Box(\forall x)A(x)$, and then we would also have $\mathcal{M}, 1.1 \Vdash_{v_0} (\forall x)A(x)$ and we do not.

$\mathcal{M}, 1 \Vdash_{v_0} (\forall x)\Box A(x)$ (Line 2) The domain $\mathcal{D}(1)$ has only q_1 as a member, so if we let w be the x -variant of v_0 such that $w(x) = v_0(q_1) = q_1$, it is enough to verify that we have $\mathcal{M}, 1 \Vdash_w \Box A(x)$. This is equivalent to verifying that we have $\mathcal{M}, 1 \Vdash_{v_0} \Box A(q_1)$, which we have.

$\mathcal{M}, 1 \Vdash_{v_0} \neg[(\forall x)\Box A(x) \supset \Box(\forall x)A(x)]$ (Line 1) because we have $\mathcal{M}, 1 \Vdash_{v_0} (\forall x)\Box A(x)$ and $\mathcal{M}, 1 \Vdash_{v_0} \neg\Box(\forall x)A(x)$.

We thus have a varying domain model in which $(\forall x)\Box A(x) \supset \Box(\forall x)A(x)$ fails at a world.

9.7 Completeness Using Maximal Consistency

We showed how maximally consistent sets could be used to prove consistency for propositional tableaus in Theorem 3.4.9, and for propositional modal tableaus in Theorem 7.6.8. The ideas extend to quantifiers quite directly. Recall that, for the modal version, besides maximal consistency we also needed *P*-completeness to take care of the ‘witnesses’ needed for possibility formulas. There is a similar idea for existential formulas, called *E*-completeness. The idea comes from Henkin’s now standard way of proving completeness for axiomatic first-order logic. To handle this the construction in the proof of Theorem 7.6.7 has two new cases added to it, and the idea behind them should be clear. We just sketch things and leave the details to you. To keep things simple, we once again state definitions and theorems and give proofs for varying domain quantified **K**, but things can easily be made to work for any of the modal logics from the Lesser Modal Cube, and for constant domains as well.

Up to now, all tableaus in this chapter began with $1 \neg\psi$ where ψ was a sentence, thus having no free variables. We must extend that. For this section tableaus can start with a finite list instead of a single item, the formulas involved can be pseudo-closed, containing parameters, and the prefixes may be something other than just 1. In the following definition the phrase *Varying Domain Tableau* occurs quite often. To keep the material readable, we use *VDT* as shorthand for this. We give the definition for **K**, but other logics from the lesser modal cube could have been used just as well.

Definition 9.7.1 (Varying Domain Tableau (VDT) Consistent) Let S be a set of prefixed pseudo-closed modal formulas. If S is finite, it is **K VDT consistent** if no (atomically) closed varying domain **K** tableau beginning with the members of S exists. If S is infinite, it is **K VDT consistent** if every finite subset is. S is *maximally K VDT consistent* if it is **K VDT consistent**, but no proper extension involving pseudo-closed formulas is. S is **K VDT inconsistent** if it is not **K VDT consistent**.

What we called *P completeness* was specified in Definition 7.6.6. We also need the similar, *E*-completeness, for quantifiers.

Definition 9.7.2 A prefixed pseudo-closed formula is *of type E* if it is one of the forms $\sigma (\exists x)\Phi(x)$ or $\sigma \neg(\forall x)\Phi(x)$. A set S of prefixed formulas is *E-complete* provided,

$$\begin{aligned} \sigma (\exists x)\Phi(x) \in S &\implies \sigma \Phi(p_\sigma) \in S \text{ for some parameter } p_\sigma, \\ \sigma \neg(\forall x)\Phi(x) \in S &\implies \sigma \neg\Phi(p_\sigma) \in S \text{ for some parameter } p_\sigma. \end{aligned}$$

For instance, Hintikka sets, Definition 9.4.1, are required to be E -complete, as well as P complete in Definition 7.5.1. Now Theorem 7.6.7 can be extended to take E -completeness into account.

Theorem 9.7.3 *Any set S of prefixed pseudo-closed formulas that is $\mathbf{K}$ VDT consistent, contains only a finite number of prefixes, and omits an infinite set of parameters, can be extended to a maximally $\mathbf{K}$ VDT consistent set that is both P -complete and E -complete.*

We only sketch the proof, which is really a straightforward elaboration of that of Theorem 7.6.7, which in turn was a tableau version of Theorem 2.4.7. The elaboration consists in modifying things to force E -completeness to hold, and we do so in a similar way to how P -completeness was ensured.

The set of all prefixed formulas is countable—assume they have been enumerated as: $\Phi_1, \Phi_2, \dots$. Likewise the set of all parameters is countable—assume it is enumerated as: $\rho_1, \rho_2, \dots$. Now, let S be a set of prefixed formulas that is $\mathbf{K}$ VDT consistent, contains a finite number of prefixes, and omits infinitely many parameters. In Fig. 9.2 we define a sequence $S_1, S_2, S_3, \dots$ of sets of prefixed formulas.

Despite what may appear complex, the ideas in the sequence definition are really quite simple. We add the formula Φ_n to S_n if the result is consistent, otherwise we don't. If Φ_n is of type P we introduce an appropriate new prefix and use it in adding a prefixed witness. And if Φ_n is of type E we introduce a new parameter and use it in adding an instantiated witness.

$$\begin{aligned}
 S_1 &= S \\
 S_{n+1} &= \begin{cases} S_n \cup \{\Psi_n\} & \text{if } S_n \cup \{\Psi_n\} \text{ is } \mathbf{K} \text{ VDT consistent and} \\ & \Psi_n \text{ is not of type } P \text{ or } E \\ S_n \cup \{\Psi_n, \sigma.k\Omega\} & \text{if } S_n \cup \{\Psi_n\} \text{ is } \mathbf{K} \text{ VDT consistent,} \\ & \Psi_n \text{ is } \sigma\Diamond\Omega, \\ & \text{and } k \text{ is the least number for which } \sigma.k \text{ is not in } S_n \\ S_n \cup \{\Psi_n, \sigma.k\neg\Omega\} & \text{if } S_n \cup \{\Psi_n\} \text{ is } \mathbf{K} \text{ VDT consistent,} \\ & \Psi_n \text{ is } \sigma\neg\Box\Omega, \\ & \text{and } k \text{ is the least number for which } \sigma.k \text{ is not in } S_n \\ S_n \cup \{\Psi_n, \sigma\Phi(\rho_i)\} & \text{if } S_n \cup \{\Psi_n\} \text{ is } \mathbf{K} \text{ VDT consistent,} \\ & \Psi_n \text{ is } \sigma(\exists x)\Phi(x), \\ & \text{and } \rho_i \text{ is the first parameter} \\ & \text{in the list of parameters that does not occur in } S_n \\ S_n \cup \{\Psi_n, \sigma\neg\Phi(\rho_i)\} & \text{if } S_n \cup \{\Psi_n\} \text{ is VDT } \mathbf{K} \text{ consistent,} \\ & \Psi_n \text{ is } \sigma\neg(\forall x)\Phi(x), \\ & \text{and } \rho_i \text{ is the first parameter} \\ & \text{in the list of parameters that does not occur in } S_n \\ S_n & \text{otherwise} \end{cases}
 \end{aligned}$$

Fig. 9.2 The Sequence $S_1, S_2, S_3, \dots$

Since $S = S_1$ contains finitely many prefixes, it follows that every S_n also does, so adding a P witness can always be done. And since S omits infinitely many parameters, so does every S_n , so adding an E witness can always be done too.

We leave it to you to verify that S_n is **K** VDT consistent for each n . And also to verify that $S_\infty = S_1 \cup S_2 \cup \dots$ is **K** VDT consistent, is maximally so, and is both P -complete and E -complete.

We now have what is needed for a second proof of completeness. We restate the theorem here for convenience, and sketch the argument.

Theorem 9.7.4 (9.5.2 Repeated) *If the sentence Φ is valid in all varying domain first-order **K** models, Φ has a tableau proof using the varying domain **K** rules.*

With all the work done (or left to the reader) the argument is now simple. As usual, it is shown in the contrapositive direction. Suppose the sentence Φ has no varying domain **K** tableau proof. Then the set $\{1 \neg\Phi\}$ is varying domain **K** tableau consistent. Now Theorem 9.7.3 applies. Extend the set to S_∞ which is maximally **K** VDT consistent, P complete, and E complete. You can check that this is a first order varying domain **K** Hintikka set, Definition 9.4.1. Then by Hintikka's Lemma, Proposition 9.4.3, it is satisfiable in a varying domain **K** model. Since it contains $1 \neg\Phi$, there is a possible world in this model at which Φ fails, and so Φ is not valid in every varying domain first-order **K** model.

References

- Fitting, M. (1996). *First-order logic and automated theorem proving* (First edition 1990). Springer-Verlag.
- Priest, G. (2008). *An introduction to non-classical logic: From if to is* (2nd ed.). *Cambridge introductions to philosophy*. First Edition published 2001. Cambridge University Press.
- Smullyan, R. M. (1968). *First-order logic* (Revised Edition, Dover Press, New York, 1994). Berlin: Springer-Verlag.

Chapter 10

First-Order Modal Axiomatics



Through all our discussions about classical and modal logics, we have examined both tableau and axiom systems. Propositionally, modal axiomatics is quite general, much more so than tableau systems. One can specify any number of modal logics axiomatically, each with its corresponding semantics, but for which no tableau systems are known. Of course, what it means to be a tableau system is somewhat open. When moving to a modal setting we added prefix machinery, in Chap. 7. In Sect. 7.7 other kinds of machinery were discussed. How much extra machinery can we add and still have something that can be considered a tableau system? But since our primary interests in this book are philosophical, relatively simple modal logics are of primary concern to us, and for these we have both axiomatics and tableaus at the propositional level.

When it comes to adding quantifiers, tableaus remain simple. The modal quantifier rules given in Chap. 9 are essentially the same as the standard classical quantifier rules, but enhanced with the additional machinery of subscripted parameters for varying domain modal logics. Axiomatics, however, becomes surprisingly complex. Completeness proofs for tableaus are fairly obvious combinations of what is done for propositional modal logics, supplemented with what is done for classical first-order logic. Axiomatic completeness proofs, however, diverge substantially. There are curious differences, for instance, depending on whether it is constant domain or varying domain logics we want. All this becomes a subject in itself, and we recommend Garson (1984), Garson (2001) for a lengthy discussion of the issues involved, and Garson (2021, Chapter 15), for a more succinct discussion. There has been work on simplifying the axiomatic complexities, see Corsi (2002) and Garson (2005), but it is not of direct relevance to our concerns here. In this chapter we briefly discuss the sources of the difficulties, and then abandon any further axiomatic treatment for the rest of this book.

10.1 A Classical First-Order Axiom System

Before bringing modal complexities into the picture, we sketch a very common axiomatization of first-order *classical* logic, so that basic quantifier machinery can be understood more easily first.

Syntactic machinery is straightforward. Instead of propositional formulas, we now use first-order ones as defined in Sect. 8.1. Even if we are interested in proving *sentences*, formulas with no free variables, proofs of sentences will involve formulas with free variables. Though we could make use of a special list of parameters, there is no particular benefit in doing so for what we are discussing now. We will just use the same list for free and bound variables.

It is certainly possible to take both quantifiers, $\forall$ (for all, the universal quantifier) and $\exists$ (there exists, the existential quantifier) as primitive. But this lengthens things, so we will take the universal quantifier as primitive and treat the existential one as defined: $(\exists x)\Phi$ abbreviates $\neg(\forall x)\neg\Phi$. We do this only here, when considering axiomatics. For tableaux, we will continue to have separate rules for each.

There are quite a number of axiomatizations possible for first-order classical logic. We present one as basic, and then discuss a few variations on it. We will get to modal issues in the next section.

Axiom Schemes

Tautologies Any instance of a classical tautology.

Universal Distributivity $(\forall x)[\Phi \supset \Psi] \supset [(\forall x)\Phi \supset (\forall x)\Psi]$.

Vacuous Quantification $\Phi \supset (\forall x)\Phi$ where x does not occur free in Φ .

Universal Instantiation $(\forall x)\Phi(x) \supset \Phi(y)$, where y is substitutable for x in $\Phi(x)$.

Rules of Inference

$$\text{Modus Ponens } \frac{\Phi \quad \Phi \supset \Psi}{\Psi}$$

$$\text{Universal Generalization } \frac{\Phi}{(\forall x)\Phi}$$

We understand a formula such as $(\forall x)R(x, y) \vee \neg(\forall x)R(x, y)$ to be a substitution instance of a propositional tautology, so it is considered to be a tautology itself, and hence an axiom. Note that free variables can occur in axioms, so $R(y, z) \supset (\forall x)R(y, z)$ is an axiom of the Vacuous Quantification type, for instance.

As usual, a *proof* is a sequence of formulas, each an axiom or following from earlier lines by a rule of inference. A *theorem* is the last line of a proof.

Here is an abbreviated example of a proof in this classical first-order axiom system.

Example 10.1.1 The formula $[(\forall x)\Phi \wedge (\forall x)\Psi] \supset (\forall x)[\Phi \wedge \Psi]$ is a theorem. Here is an abbreviated proof. We have left out steps that are easily justified using propositional logic.

- | | |
|---|-----------------------------|
| 1. $\Phi \supset [\Psi \supset (\Phi \wedge \Psi)]$ | Tautology |
| 2. $(\forall x)\{\Phi \supset [\Psi \supset (\Phi \wedge \Psi)]\}$ | Univ. Gen. on 1 |
| 3. $(\forall x)\Phi \supset (\forall x)[\Psi \supset (\Phi \wedge \Psi)]$ | Univ. Distrib. and 2 |
| 4. $(\forall x)[\Psi \supset (\Phi \wedge \Psi)] \supset [(\forall x)\Psi \supset (\forall x)(\Phi \wedge \Psi)]$ | Univ. Distrib. |
| 5. $(\forall x)\Phi \supset [(\forall x)\Psi \supset (\forall x)[\Phi \wedge \Psi]]$ | from 3, 4 using prop. logic |
| 6. $[(\forall x)\Phi \wedge (\forall x)\Psi] \supset (\forall x)[\Phi \wedge \Psi]$ | from 5 using prop. logic |

Exercises

Exercise 10.1.1 Give axiomatic proofs of the following.

- $(\forall x)[\Phi(x) \wedge \Psi(x)] \supset [(\forall x)\Phi(x) \wedge (\forall x)\Psi(x)]$.
(We now have provability of $(\forall x)[\Phi(x) \wedge \Psi(x)] \equiv [(\forall x)\Phi(x) \wedge (\forall x)\Psi(x)]$.)
- If x does not occur free in Φ , $[\Phi \vee (\forall x)\Psi(x)] \supset (\forall x)[\Phi \vee \Psi(x)]$.
- If x does not occur free in Φ , $(\forall x)[\Phi \vee \Psi(x)] \supset [\Phi \vee (\forall x)\Psi(x)]$.
Hint: $[A \supset (B \vee C)] \equiv [(A \wedge \neg B) \supset C]$.
- If x does not occur free in Φ , $[\Phi \wedge (\exists x)\Psi(x)] \equiv (\exists x)[\Phi \wedge \Psi(x)]$. Recall that $(\exists x)$ abbreviates $\neg(\forall x)\neg$.
- $(\exists x)[\Phi(x) \supset (\forall x)\Phi(x)]$. Hint: $\neg(P \supset Q) \equiv (P \wedge \neg Q)$.
- $[(\forall x)\Phi(x) \wedge (\exists x)\Psi(x)] \supset (\exists x)[\Phi(x) \wedge \Psi(x)]$.

10.2 So What Are the Problems?

To get an axiomatized first-order *modal* logic, a natural approach would be to simply combine a propositional modal axiomatization with the standard first-order classical axiom system we have just seen. Doing this gives us an axiomatically formulated modal logic with quantifiers and, as such, is probably the first version anyone would think of. The natural question then would be, what is the corresponding semantics? In Chap. 8 two primary versions of first-order modal Kripke models were presented: varying domain and constant domain. Which of these two would the axiomatization just described be sound and complete with respect to? It turns out to be neither of them!

Suppose we try examining the approach just sketched, using the modal logic **K** as a representative modal logic. We employ a first-order modal language, as set out in Sect. 8.1. As our first-order axiomatic machinery we use what is given in Sect. 10.1,

and as modal axioms and rules we use those of Sect. 6.1. We simply combine it all. Here is a somewhat abbreviated proof in the resulting system.

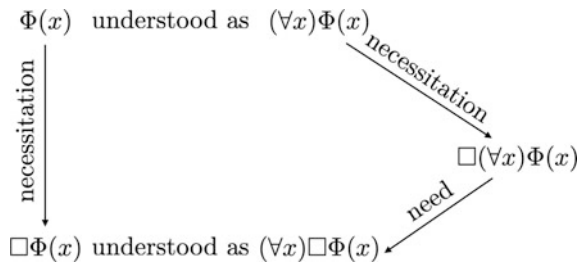
- | | |
|---|-----------------------|
| 1. $(\forall x)\Phi(x) \supset \Phi(x)$ | Univ. Inst. |
| 2. $\Box[(\forall x)\Phi(x) \supset \Phi(x)]$ | Nec. Rule on 1 |
| 3. $\Box(\forall x)\Phi(x) \supset \Box\Phi(x)$ | K Schema and 2 |
| 4. $(\forall x)[\Box(\forall x)\Phi(x) \supset \Box\Phi(x)]$ | Univ. Gen. on 3 |
| 5. $(\forall x)\Box(\forall x)\Phi(x) \supset (\forall x)\Box\Phi(x)$ | Univ. Dist. and 4 |
| 6. $\Box(\forall x)\Phi(x) \supset (\forall x)\Box(\forall x)\Phi(x)$ | Vac. Quant. |
| 7. $\Box(\forall x)\Phi(x) \supset (\forall x)\Box\Phi(x)$ | from 5 and 6 |

We thus have a proof of the Converse Barcan Formula, Definition 8.10.2. That means a semantics corresponding to our axiom system for first-order **K** should meet a monotonicity condition, by Proposition 8.10.6. Then the appropriate semantic counterpart for our axiomatization cannot be the full varying domain semantics.

On the other hand, the Barcan Formula itself is not provable. We leave it to you to check that every one of our axioms is valid in the varying domain model shown in Example 8.10.3, and validity is preserved in that model under all of our rules of inference, but an instance of the Barcan formula fails in it. Using Proposition 8.10.8, a semantic counterpart to our axiomatically formulated logic should not require anti-monotonicity. This rules out a constant domain semantics.

In fact, it is possible to show a completeness theorem stating that the axiomatically provable formulas are those valid in exactly the varying domain models that are monotonic, provided a special restriction on validity in models is imposed: X is valid in the first-order axiomatization for **K** sketched above provided that for every varying domain model $\langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ that is monotonic (Definition 8.10.4), and for every $\Gamma \in \mathcal{G}$, we have $\Gamma \Vdash_v X$ for every valuation v that maps the free variables of X to $\mathcal{D}(\Gamma)$. We will not follow this up any further here. But the fundamental fact is, we have not axiomatized either constant domain semantics or varying domain semantics, but rather a monotonic version intermediate between the two.

As Kripke originally discussed in his fundamental paper, Kripke (1963), this monotonic outcome has a plausible source. In classical first-order proofs, free variables are understood semantically as if they were universally quantified. That is, $\Phi(x)$ and $(\forall x)\Phi(x)$ are understood to behave alike. But this brings on some problems when we move to a modal context. Consider the following diagram.



Suppose we have $\Phi(x)$ as a line in a proof. We are allowed to apply the Necessitation Rule to it, getting $\Box\Phi(x)$. But our informal understanding of $\Phi(x)$, containing a free variable, is that it behaves like $(\forall x)\Phi(x)$. If we apply necessitation to our understanding of the formula we get $\Box(\forall x)\Phi(x)$, while our understanding of $\Box\Phi(x)$, again containing a free variable, is that it behaves like $(\forall x)\Box\Phi(x)$. There is a mismatch here, which could be addressed if we could convert $\Box(\forall x)\Phi(x)$ into $(\forall x)\Box\Phi(x)$. But this is just what the Converse Barcan Formula does.

This suggests we should not be surprised to find difficulties when necessitation is applied to formulas containing free variables. And it suggests that we might avoid ambiguity by being more explicit about how we are understanding free variable occurrences. In the next two sections we briefly sketch how constant and varying domain modal semantics can be handled axiomatically. One of the two versions involves getting rid of formulas with free variables in proofs entirely! After this, we abandon axiomatic approaches altogether and concentrate on tableaux.

10.3 Constant Domain Systems

We know that the Converse Barcan Formula and the Barcan formula have connections with monotonicity and anti-monotonicity of frames. We have just seen that if we combine the propositional modal axiomatics from Chap. 6 with the classical axiomatics of Sect. 10.1, the Converse Barcan Formula becomes provable, though the Barcan formula is not, at least for a logic based on **K**. Suppose we simply add the Barcan Formula as an additional axiom scheme. It turns out that this is a successful axiomatization for a constant domain version of modal logic **L**, for each **L** from the Modal Cube. We do not prove this here. Proofs can be found in Hughes and Cresswell (1996) and Garson (1984, 2001), for instance. There are differences that turn up in their axiomatization of quantifiers, but this is more than we can cover in our brief summary here.

Example 10.3.1 The following is a proof using the constant domain axiomatization for **T** of

$$[(\forall x)\Box\Box\Phi(x) \wedge \Diamond(\exists x)\Psi(x)] \supset \Diamond(\exists x)[\Phi(x) \wedge \Psi(x)].$$

As usual, it is much abbreviated.

- | | |
|---|-----------------------------------|
| 1. $\Box\Box\Phi(x) \supset \Box\Phi(x)$ | T axiom |
| 2. $(\forall x)\Box\Box\Phi(x) \supset (\forall x)\Box\Phi(x)$ | 1, Univ. Gen. and Univ. Dist. |
| 3. $(\forall x)\Box\Phi(x) \supset \Box(\forall x)\Phi(x)$ | Barcan formula |
| 4. $(\forall x)\Box\Box\Phi(x) \supset \Box(\forall x)\Phi(x)$ | from 2,3 |
| 5. $[(\forall x)\Box\Box\Phi(x) \wedge \Diamond(\exists x)\Psi(x)] \supset$
$\quad [\Box(\forall x)\Phi(x) \wedge \Diamond(\exists x)\Psi(x)]$ | from 4 using classical reasoning |
| 6. $[\Box(\forall x)\Phi(x) \wedge \Diamond(\exists x)\Psi(x)] \supset$
$\quad \Diamond[(\forall x)\Phi(x) \wedge (\exists x)\Psi(x)]$ | theorem of propositional K |
| 7. $[(\forall x)\Phi(x) \wedge (\exists x)\Psi(x)] \supset$
$\quad (\exists x)[\Phi(x) \wedge \Psi(x)]$ | Exercise 10.1.1 |
| 8. $\Diamond[(\forall x)\Phi(x) \wedge (\exists x)\Psi(x)] \supset$
$\quad \Diamond(\exists x)[\Phi(x) \wedge \Psi(x)]$ | from 7 by Exercise 6.1.1 |
| 9. $[(\forall x)\Box\Box\Phi(x) \wedge \Diamond(\exists x)\Psi(x)] \supset$
$\quad \Diamond(\exists x)[\Phi(x) \wedge \Psi(x)]$ | from 5, 6, 8 |

Validity of the Converse Barcan Formula corresponds to monotonicity of a skeleton; validity of the Barcan Formula to anti-monotonicity. If the accessibility relation of a skeleton is symmetric, one would expect Barcan and Converse Barcan to stand or fall together. (Admittedly, this is somewhat imprecise.) Stating things in the context of the modal axiomatic system from Sect. 10.2, since it can prove the Converse Barcan formula, we might expect that if the underlying modal logic is at least as strong as the symmetric logic **B**, then the Barcan Formula should also be provable. This is indeed the case, as Example 10.3.2 shows, giving a proof that derives from Prior.

Example 10.3.2 The following is a proof of the Barcan formula in a first-order axiomatization of **B**, where the axiomatization follows the lines of that for **K** from Sect. 10.2. We use axiom schema *B* in the form $P \supset \Box\Diamond P$ for line 9, and in the equivalent version $\Diamond\Box P \supset P$ for line 5.

- | | |
|--|------------------------------------|
| 1. $(\forall x)\Box\Phi(x) \supset \Box\Phi(x)$ | Univ. Inst. |
| 2. $\Box[(\forall x)\Box\Phi(x) \supset \Box\Phi(x)]$ | Nec. rule on 1 |
| 3. $(\forall x)\Box[(\forall x)\Box\Phi(x) \supset \Box\Phi(x)]$ | Univ. Gen. rule on 2 |
| 4. $(\forall x)[\Diamond(\forall x)\Box\Phi(x) \supset \Diamond\Box\Phi(x)]$ | from 3 using Exercise 6.1.3 part 2 |
| 5. $(\forall x)[\Diamond(\forall x)\Box\Phi(x) \supset \Phi(x)]$ | from 4 using <i>B</i> axiom |
| 6. $(\forall x)\Diamond(\forall x)\Box\Phi(x) \supset (\forall x)\Phi(x)$ | from 5 using Univ. Dist. |
| 7. $\Diamond(\forall x)\Box\Phi(x) \supset (\forall x)\Phi(x)$ | from 6 using Vac. Quant. |
| 8. $\Box\Diamond(\forall x)\Box\Phi(x) \supset \Box(\forall x)\Phi(x)$ | from 7 using Reg. Rule |
| 9. $(\forall x)\Box\Phi(x) \supset \Box(\forall x)\Phi(x)$ | from 8 using <i>B</i> axiom |

Exercises

Exercise 10.3.1 Give constant domain axiomatic **K** proofs of the following.

1. $\Diamond(\forall x)A(x) \supset (\forall x)\Diamond A(x)$.
2. $(\exists x)\Box A(x) \supset \Box(\exists x)A(x)$.
3. $[\Diamond(\forall x)A(x) \wedge \Box(\exists x)B(x)] \supset (\exists x)\Diamond[A(x) \wedge B(x)]$.

Exercise 10.3.2 Give constant domain axiomatic **S4** proofs of the following.

1. $\Box(\forall x)\Box A(x) \equiv (\forall x)\Box A(x)$.
2. $\Box(\exists x)\Box A(x) \equiv (\exists x)\Box A(x)$.

Exercise 10.3.3 Give a constant domain axiomatic **S5** proof of the following. Since **S5** includes *B* as an axiom scheme, you don't need to explicitly assume the Barcan or Converse Barcan formulas, though you can do so if you wish.

$$\Box(\forall x)A(x) \equiv \Diamond(\forall x)\Box A(x)$$

10.4 Varying Domain Systems

In the most common axiomatizations of first-order classical logic, Universal Instantiation is taken as an axiom scheme, and free variables are allowed in proofs. But *Universal Instantiation is not a sound rule when using the varying domain modal semantics of Sect. 8.7*. The reasons are quite straightforward. Valuations in modal models assign members of the domain of the *model* to free variables, while quantifiers quantify over domains of *particular worlds*. Thus, in a varying domain model $\mathcal{M}$ we might have $\mathcal{M}, \Gamma \Vdash_v (\forall x)\Phi(x)$, because $\Phi(x)$ is true at Γ when x is given any value in $\mathcal{D}(\Gamma)$, the domain of world Γ . But at the same time we might also have $\mathcal{M}, \Gamma \not\Vdash_v \Phi(y)$ because v assigns to the free variable y something in the domain of the model $\mathcal{M}$ that is not in $\mathcal{D}(\Gamma)$, and for which $\Phi(y)$ fails at Γ . Thus $(\forall x)\Phi(x) \supset \Phi(y)$ can fail to be true in a varying domain model.

There is a plausible solution to this problem. In $(\forall x)\Phi(x) \supset \Phi(y)$, the quantifier ranges over a set that can be smaller than the set of values allowed for y . But, the values assignable to y would be restricted if y were quantified too. We could use the following instead of the version of Universal Instantiation we adopted earlier: $(\forall y)[(\forall x)\Phi(x) \supset \Phi(y)]$. This formula *is* valid in the varying domain modal context. Classically the two versions are essentially interchangeable, semantically. Modally they are not, and care must be exercised.

Kripke (1963) proposed the first varying domain axiomatization. It follows the non-modal lead of Quine (1940) in allowing only closed formulas to appear in proofs. In particular, in place of $(\forall x)\Phi(x) \supset \Phi(y)$ one uses $(\forall y)[(\forall x)\Phi(x) \supset \Phi(y)]$ for instantiating universally quantified formulas. Otherwise it builds on the classical axiom system of Sect. 10.1. But free variables were allowed in that axiom

system, so Kripke required that all axioms be *closures* of those schemas, but with a very special definition of closure. The following is a quote from Kripke (1963).

If A is a formula containing free variables, we define a *closure* of A to be any formula without free variables obtained by prefixing universal quantifiers and necessity signs, in any order, to A .

For instance, if $P(x, y, z)$ and $Q(x, y, z)$ are atomic then $(\forall x)[P(x, y, z) \supset Q(x, y, z)] \supset [(\forall x)P(x, y, z) \supset (\forall x)Q(x, y, z)]$ is not an axiom, but the following all are (the list of possibilities is not complete).

$$\begin{aligned} & (\forall y)(\forall z)\{(\forall x)[P(x, y, z) \supset Q(x, y, z)] \supset [(\forall x)P(x, y, z) \supset (\forall x)Q(x, y, z)]\} \\ & (\forall y)\Box\Box(\forall z)\{(\forall x)[P(x, y, z) \supset Q(x, y, z)] \supset [(\forall x)P(x, y, z) \supset (\forall x)Q(x, y, z)]\} \\ & \Box(\forall y)\Box(\forall z)\{(\forall x)[P(x, y, z) \supset Q(x, y, z)] \supset [(\forall x)P(x, y, z) \supset (\forall x)Q(x, y, z)]\} \\ & \Box(\forall y)(\forall z)\Box\Box\{(\forall x)[P(x, y, z) \supset Q(x, y, z)] \supset [(\forall x)P(x, y, z) \supset (\forall x)Q(x, y, z)]\} \end{aligned}$$

This use of closures not only gives us formulas with no free variables as axioms, but provides an alternative to the Necessitation Rule, avoiding the clash that arises at the interface of that rule and universal instantiation, as discussed in Sect. 10.2.

Actually, Kripke omitted one more essential axiom. Permutation is a validity of classical logic, $(\forall x)(\forall y)\Phi \supset (\forall y)(\forall x)\Phi$, but as was shown in Fine (1983), it is not provable in Kripke's system as stated. When added, we have a sound and complete axiom system for modal logic **L**, once the modal axioms for **L** are added.

We note that Hughes and Cresswell (1996), starting on p. 304, gives an alternate axiomatization which is simpler in several respects. Likewise, Fitting (2020) provides an alternate way of thinking about Kripke's axiomatization. Completeness proofs can be found in Hughes and Cresswell (1996), Garson (1984), and Garson (2001).

Example 10.4.1 Here is a varying domain proof using the **T** rules of the sentence $(\forall x)\Box[\Phi(x) \wedge \Psi(x)] \supset (\forall y)\Phi(y)$. As usual, we present a somewhat abbreviated version.

- | | |
|---|---------------------------|
| 1. $(\forall y)\{[\Phi(y) \wedge \Psi(y)] \supset \Phi(y)\}$ | closure of tautology |
| 2. $(\forall y)\{\Box[\Phi(y) \wedge \Psi(y)] \supset [\Phi(y) \wedge \Psi(y)]\}$ | closure of T axiom |
| 3. $(\forall y)\{\Box[\Phi(y) \wedge \Psi(y)] \supset \Phi(y)\}$ | from 1, 2 |
| 4. $(\forall y)\Box[\Phi(y) \wedge \Psi(y)] \supset (\forall y)\Phi(y)$ | from 3 by Univ. Dist. |
| 5. $(\forall y)\{(\forall x)\Box[\Phi(x) \wedge \Psi(x)] \supset \Box[\Phi(y) \wedge \Psi(y)]\}$ | Univ. Inst. |
| 6. $(\forall y)(\forall x)\Box[\Phi(x) \wedge \Psi(x)] \supset (\forall y)\Box[\Phi(y) \wedge \Psi(y)]$ | from 5 by Univ. Dist. |
| 7. $(\forall y)(\forall x)\Box[\Phi(x) \wedge \Psi(x)] \supset (\forall y)\Phi(y)$ | from 4 and 6 |
| 8. $(\forall x)\Box[\Phi(x) \wedge \Psi(x)] \supset (\forall y)\Phi(y)$ | from 7 by Vac. Quant. |

Exercises

Exercise 10.4.1 In Example 10.4.1, give a detailed argument, using Kripke's varying domain axiomatization and Fine's addition, that line 3 follows from lines 1 and 2.

Exercise 10.4.2 Explicitly give a varying domain **K** model in which $(\forall x)P(x) \supset P(y)$ is not valid, where P is a one place relation symbol.

Exercise 10.4.3 Give a varying domain axiomatic **K** proof of the following.

$$[\Diamond(\forall x)A(x) \wedge \Box(\exists x)B(x)] \supset \Diamond(\exists x)[A(x) \wedge B(x)]$$

Exercise 10.4.4 Give a varying domain axiomatic **D** proof of the following.

$$\Box(\forall x)[A(x) \vee B(x)] \supset [\Diamond(\forall x)A(x) \vee \Diamond(\exists x)B(x)]$$

Exercise 10.4.5 Show soundness for varying domain axiomatic **K**.

References

- Corsi, G. (2002). A unified completeness theorem for quantified modal logics. *Journal of Symbolic Logic*, 67(4), 1483–1510.
- Fine, K. (1983). The permutation principle in quantificational logic. *Journal of Philosophical Logic*, 12, 33–37.
- Fitting, M. (2020). De re, de dicto, and binding modalities. In F. Liu, H. Ono, & J. Yu (Eds.), *Knowledge, Proof and Dynamics: The Fourth Asian Workshop on Philosophical Logic* (pp. 147–171). Logic in Asia: Studia Logica Library. Singapore: Springer.
- Gabbay, D. M. & Guenther, F. (Eds.). (1983–1989). *Handbook of philosophical logic*. Kluwer. Four volumes. Dordrecht: Kluwer.
- Gabbay, D. M. & Guenther, F. (Eds.). (2001 to present). *Handbook of philosophical logic* (2nd ed.). Springer. Multiple volumes. Dordrecht: Kluwer.
- Garson, J. W. (1984). Quantification in modal logic (Chap. 5, Vol. 2, pp. 249–307). In *Gabbay & Guenther*, 1983–1989.
- Garson, J. W. (2001). Quantification in modal logic. (Chap. 3, pp. 267–323). In *Gabbay & Guenther*, 2001 to present. Springer.
- Garson, J. W. (2005). Unifying quantified modal logic. *Journal of Philosophical Logic*, 34, 621–649.
- Garson, J. W. (2021). Modal logic. In E. N. Zalta (Ed.), *The Stanford encyclopedia of philosophy* (Summer 2021 Edition). Stanford University.
- Hughes, G. E. & Cresswell, M. J. (1968). *A new introduction to modal logic*. London: Routledge.
- Kripke, S. (1963). Semantical considerations on modal logic. *Acta Philosophica Fennica*, 16, 83–94.
- Quine, W. V. O. (1940). *Mathematical logic* (2nd ed., Revised, 1951). Harvard University Press.

Part IV

Equality and Existence

In a modal model free variables range over the things (or objects) of that model. What is a thing? It can be anything. Something concrete like tables and chairs and beer mugs,¹ or abstract like the concepts of truth and beauty. The semantics we have been presenting is meant to be very general, and whatever can be said that is always correct is a logical truth. But this is exactly why the semantics is useful to us.

What else might we want to say? What kinds of meaningful statements about entirely arbitrary things are there besides the universal truths? Probably among the most basic of truths is that we have such-and-such an object (existence) or that we have only one such object (uniqueness). These are concepts that make sense universally, but sometimes apply truly and sometimes don't. Existence, more formally, entails we should have machinery for saying that at a possible world the value of a free variable x is some thing that exists at that world. Equality further entails we should have machinery for saying, of free variables x and y , whether the things that are their values are the same or different. That is, whether they are equal or not. These topics have relationships to each other, and it is these, existence and equality, that we examine next.

¹ The mathematician David Hilbert once said, when discussing the abstractness of geometry, "One must be able to say at all times—instead of points, straight lines, and planes—tables, chairs, and beer mugs." Reported at the beginning of Chapter VIII in *Hilbert*, by Constance Reid, Springer (1996).

Chapter 11

Equality



11.1 Classical Background

When we say that two people own the same car, we might mean either they own the same *type* of car, e.g., a Honda, or they are joint owners of a single car. In the former case, *same* means *qualitative identity* or *equivalence*; in the latter case, it means *quantitative or numerical identity*, or, as we shall call it, *equality*. Equality is actually a special case of an *equivalence relation*. We have already seen equivalence relations, in Sect. 5.4 when we introduced the logics making up the modal cube. There we were interested in the logics we got by using combinations of properties that together make up an equivalence relation. Now we are interested in the relation itself.

Definition 11.1.1 (Equivalence Relation) A binary relation $\sim$ on a set of objects S is an *equivalence relation* if for all $x, y \in S$

1. $x \sim x$, i.e., $\sim$ is *reflexive*.
2. $x \sim y$ implies $y \sim x$, i.e., $\sim$ is *symmetric*.
3. $x \sim y$ and $y \sim z$ together imply $x \sim z$, i.e., $\sim$ is *transitive*.

An equivalence relation $\sim$ on S determines a *partition* of the set S into disjoint subsets called *equivalence classes*. Each element in an equivalence class bears the relation $\sim$ to every element in that class, and to none outside—its elements are equivalent exactly to one another. For example, if the relation is *same manufacturer* and we are talking about the set S of cars, the same type of car equivalence relation partitions S into the set of Hondas, the set of Toyotas, of Fords, of Fiats, and so forth. *Equality* is also an equivalence relation: it is the *smallest* equivalence relation, so that each one of the equivalence classes is a singleton, i.e., each contains one element.

Definition 11.1.2 (Equivalence Class) Let $\sim$ be an equivalence relation on a set S . For each $p \in S$, by $\bar{p}$ we mean $\{q \in S \mid q \sim p\}$.

Thus $\bar{p}$ is the set of all members of S that are equivalent to p . This is called the *equivalence class of p* . The following establishes the important and fundamental properties that all equivalence classes have.

Proposition 11.1.3 *Let $\sim$ be an equivalence relation on a set S .*

1. *Every member of S is in some equivalence class. In particular, $p \in \bar{p}$.*
2. *No member of S is in more than one equivalence class. In particular, if $\bar{p}$ and $\bar{q}$ have any member in common, then $\bar{p} = \bar{q}$.*
3. *$p \sim q$ if and only if $\bar{p} = \bar{q}$.*

Proof Item 1 is the easiest. Since equivalence is reflexive, $p \sim p$, and it follows that $p \in \bar{p}$.

For item 2, assume $r \in \bar{p}$ and $r \in \bar{q}$. We show $\bar{p} \subseteq \bar{q}$; the converse inclusion is similar. So, suppose $x \in \bar{p}$; we must show $x \in \bar{q}$. Since $r \in \bar{p}$, $r \sim p$, and since $r \in \bar{q}$, $r \sim q$. Since $r \sim p$, by symmetry $p \sim r$. Then since also $r \sim q$, by transitivity $p \sim q$. Now, $x \in \bar{p}$, so $x \sim p$. Since $p \sim q$, by transitivity again, $x \sim q$, so $x \in \bar{q}$.

Finally, for item 3. In one direction, suppose $\bar{p} = \bar{q}$. By item 1, $p \in \bar{p}$, so $p \in \bar{q}$, and hence $p \sim q$. In the other direction, suppose $p \sim q$. Then $p \in \bar{q}$ and since $p \in \bar{p}$, the equivalence classes $\bar{p}$ and $\bar{q}$ have a member in common, hence $\bar{p} = \bar{q}$ by item 2.

In classical first-order logic with equality, we have a two-place relation $x = y$ customarily governed by two conditions. The first,

$$x = x, \tag{11.1}$$

explicitly gives us reflexivity. The second is commonly called *Leibniz's Law*:

$$(x = y) \supset (\Phi(x) \supset \Phi(y)). \tag{11.2}$$

In this, $\Phi(y)$ is meant to be $\Phi(x)$ with some, not necessarily all, occurrences of x replaced with occurrences of y , and it is allowed that other variables might be present in $\Phi(x)$ besides x .

This yields both symmetry and transitivity by using judicious choices for the schematic letter Φ . We leave this as an exercise. This assures that $=$ is an equivalence relation.

Whether equality has been captured by these conditions is open to question. In second-order logic, equality is definable: $x = y$ is taken to mean $(\forall \Phi)(\Phi(x) \equiv \Phi(y))$. That is, x is equal to y iff they have *all* their properties in common. In first-order logic, however, we cannot speak of *all properties*. Formula (11.2) is really a schema with Φ replaceable by the available predicates of the language. And the available predicates might not be sufficient to discern all choices of distinct objects.

Suppose, for example, that the only predicates in the theory are “ x is a dog” and “ x is a bird.” Suppose further that we have as an axiom that no dog is a bird and conversely. This theory cannot distinguish among birds because there is no property

any one bird has that any other bird lacks. (Likewise, it cannot distinguish among dogs.) Since the theory cannot distinguish among objects that belong to the same species, it is possible to consistently interpret $x = y$ as the equivalence relation *x is an element of the same species as y* . And this equivalence relation is *not* equality. The problem is the lack of expressiveness of our language—there are not enough predicates.

All of the usual principles governing equality are obtainable without necessarily interpreting $=$ as equality. In particular *The Indiscernibility of Identicals* is valid in all classical first-order models:

$$(x = y) \supset (\Phi(x) \equiv \Phi(y)). \quad (11.3)$$

It is enough to interpret $=$ as an equivalence relation that cannot distinguish between two objects, x and y , in the domain of the model that happen to share all properties *definable* in the model. Since such properties may not be *all* properties, interpreting $=$ to be something less than full equality can suffice.

There are first-order models of classical logic in which $=$ is interpreted as standard equality. This is, of course, the normal or intended interpretation; accordingly, a model in which $=$ is taken to be equality is said to be a *normal* model. But, as we have just noted, there are first-order models of classical logic in which $=$ is interpreted in a way other than standard equality. So, we cannot claim that we have captured equality completely with the two conditions (11.1) and (11.2). But we have, in fact, captured equality in the sense that a formula is true in all normal models if and only if it is true in all models allowing the kind of non-standard reading of $=$ discussed above. We give an example of how to convert a non-standard model into a normal one in Sect. 11.9.

We carry over to the modal setting these ideas about equality from classical logic. But the subject of equality has caused much grief for modal logic. Many of the problems that have been posed for modal logic, and which historically have struck at the heart of the coherence of modal logic, stem from apparent violations of the Indiscernibility of Identicals. We will discuss some of the problematic cases.

We begin with one of the most famous puzzles concerning equality and modal logic.

Exercises

Exercise 11.1.1 Show that $=$ is transitive and symmetric, assuming (11.1) and (11.2).

11.2 Frege's Puzzle

The need for an equality predicate in classical logic is clear. But further, in modal logic we want to distinguish necessary equalities, e.g., that π is the ratio of the circumference of a circle to its diameter, from merely contingent ones, e.g., that George Washington is the first American president. Various philosophical puzzles, however, have stood in the way of this clear-sighted appraisal.

Perhaps the best known of the puzzles was introduced by Frege (1892):

Equality gives rise to challenging questions which are not altogether easy to answer. ... $a = a$ and $a = b$ are obviously statements of differing cognitive value; $a = a$ holds *a priori* and, according to Kant, is to be labeled analytic, while statements of the form $a = b$ often contain very valuable extensions of our knowledge and cannot always be established *a priori*. The discovery that the rising sun is not new every morning, but always the same, was one of the most fertile astronomical discoveries. Even today the identification of a small planet or a comet is not always a matter of course. Now if we were to regard equality as a relation between that which the names ' a ' and ' b ' designate, it would seem that $a = b$ could not differ from $a = a$ (i.e. provided $a = b$ is true). A relation would thereby be expressed of a thing to itself but to no other thing. (pp. 56–57)

This, of course, is the well-known morning star/evening star puzzle. If we go far enough back in time, we reach a point when it was not known that the morning star and the evening star are the same. Still, to say that a and b are equal is to say they are the same thing, and consequently whatever is true of a is true of b , and conversely. Now, the morning star and the evening star are, in fact, equal. And certainly, the ancients knew the morning star was the morning star. Then it should follow that the ancients knew the evening star was the morning star. But they didn't.

Wittgenstein (1922) succumbed to this puzzle completely: "...to say of *two* things that they are identical is nonsense, and to say of *one* thing that it is identical with itself is to say nothing at all." (§5.5303) He persuaded himself that there were no informative equalities, and so no need for the equality predicate:

Identity of object I express by identity of sign, and not by using a sign for identity.
Difference of objects I express by difference of signs. (§5.53)

But Wittgenstein's notation is inadequate. We cannot prove the Pythagorean Theorem in his symbolism. We cannot even express it!

Frege (1879) succumbed only partially. He was persuaded, by the puzzle, that equality could not relate objects. But he appreciated the utility of equality, and attempted to account for the informative cases by taking equality to relate the names of the objects they designate. This solution is unsatisfactory and Frege came to abandon it. Originally he failed to distinguish clearly between what one is talking about and what one is saying about it, a distinction elaborated in his celebrated sense/reference theory announced in Frege (1892). (See Mendelsohn (1982) for further discussion.)

Frege (1892) diagnosed the puzzle as resting on the false assimilation of sense [*Sinn*] and reference [*Bedeutung*]. The two names "morning star" and "evening star" refer to the same object—the planet Venus—but do so in different ways, and

so differ in sense. Accordingly, the proposition that the morning star is equal to the morning star differs from the proposition that the morning star is equal to the evening star. It should come as no surprise that the ancients knew the one proposition to be true without knowing the other to be true.

But Frege's solution provides no relief for modal versions of the puzzle. In fact, it seems to exacerbate them. The simplest sentence saying one can substitute equals for equals in a modal context is

$$(x = y) \supset \Box(x = y). \quad (11.4)$$

If we interpret $\Box$ as "the ancients knew," and a and b as the morning and evening star respectively, then $(a = b) \supset \Box(a = b)$ is an instance of (11.4). Since $a = b$ is factually true, *Modus Ponens* brings us to the problem immediately. A number of philosophers and logicians have found this result unpalatable. As a result, they have rejected (11.4), and since (11.4) is valid in first-order modal logic—we will actually demonstrate (11.4) in Sect. 11.3—they have rejected quantified modal logic as well.

In the process of providing a reasonably effective solution to the puzzle about equality for *classical* logic, Frege's sense/reference solution renders equality precarious in *modal* logic, for it provides every reason to reject (11.4). Even though the objects designated by a and b are one and the same, these names need not have the same sense. The proposition that $a = a$, which is trivially true, would then be distinct from the proposition that $a = b$, which is not. And so, the fact that $a = b$ is true would provide no reasonable assurance that $\Box(a = b)$ is true.

Frege held that when a name occurs inside a modal context, it is used to speak about its sense, not its designation. So, to say that the ancients knew the morning star is the evening star is not to speak about the celestial body nor to ascribe a property to that celestial body. It is to speak about the senses of the words, or, alternatively, about the proposition expressed by $a = b$. In other words, Frege's understanding of the modal operator is *de dicto*. He has no *de re* reading. (Recall, these were discussed in Sect. 8.3.) But it is the *de re* reading that is crucial for first-order modal logic, and, in particular, for the validation of (11.4).¹ By effectively precluding a *de re* reading for modal statements, his approach to the problem of equality has (in yet another way) impeded the development of modal logic.

Russell (1905) provides the most satisfying solution to Frege's puzzle, although the full impact of his solution was not understood until a half-century later. Speaking a bit anachronistically, Russell held that for genuine proper names, the *Sinn* is the *Bedeutung*, but for definite descriptions and names that are disguised definite descriptions, the two are distinct. More importantly, he introduced a scope distinction for definite descriptions that, when suitably elaborated by (Arthur, not

¹ With his *Sinn/Bedeutung* distinction, Frege—despite the attitude we described earlier in Sect. 4.2—has long been thought to provide the philosophical underpinnings for a semantics for modal logic (as was carried out, for example, by his most famous pupil Carnap (1947)). Ironically, Quine's criticisms of modal logic, as detailed in Chap. 8, appear to be heavily influenced by Frege's semantics for modal contexts.

Raymond) Smullyan (1948), enabled him to formalize the *de re/de dicto* distinction. This is the path we follow in this book. On our view, the puzzle about informative identities arises because of the interplay of modal operators and singular terms. Our solution to the puzzle will await the introduction of predicate abstraction machinery, starting in Part V. But there is no puzzle at this stage about (11.4) which contains only *variables*. If x and y are the very same object, and x has the property of being necessarily identical with x , then y has that very same property. The puzzle arises when we replace these variables with singular terms.

Exercises

Exercise 11.2.1 Suppose a and b are not names or definite descriptions, but demonstratives such as “this” and “that.” Can $a = a$ differ in cognitive value from $a = b$? Discuss.

Exercise 11.2.2 “Frege’s puzzle about equality is not really about equality. It is about meaning.” Argue for or against this claim.

Exercise 11.2.3 Wittgenstein suggested eliminating the equality sign. How impaired would the resulting language be, if impaired at all?

11.3 The Indiscernibility of Identicals

The Indiscernibility of Identicals, (11.3), is among the most fundamental of logical principles. To deny it is to suppose that $x = y$ and yet that x has a property y lacks, or conversely. Recall, x and y are not *names*, they are variables which have *objects* as values. Denying the Indiscernibility of Identicals for objects is to suppose that the object x both has and lacks a particular property. In other words, *The Indiscernibility of Identicals* is on a par with the venerable *Law of Noncontradiction*, $\neg(\exists x)(\Phi(x) \wedge \neg\Phi(x))$.

Closely related to the Indiscernibility of Identicals, (11.3), and frequently conflated with it, is a *Substitution Principle*:

If a and b denote the same thing, then $\Phi(a)$ and $\Phi(b)$ have
the same truth value, (11.5)

where a and b are any two singular terms. Unlike (11.3), the *Substitution Principle* is easily shown to be false; the morning star/evening star example we discussed in Sect. 11.2 shows this. But counterexamples to the substitution principle in no way provide counterexamples to the Indiscernibility of Identicals. On the contrary, counterexamples like the one just described are due to the behavior of names and

descriptions. However (11.3) holds in a first-order language (like the one we are considering) which contains neither names nor descriptions, but only variables.

Take Φ in (11.3) to be $\Box(x =)$, giving us

$$(x = y) \supset (\Box(x = x) \equiv \Box(x = y)). \quad (11.6)$$

Since $x = x$ is valid, by *Necessitation* we also have $\Box(x = x)$. Equation (11.6), therefore, yields

$$(x = y) \supset \Box(x = y), \quad (11.7)$$

just as we promised in Sect. 11.2. (For a semantical version of this argument, see Example 11.4.2.)

Formula (11.7) appears to fly in the face of the clear intuition that some identities are necessary while others are only contingent. Frege's original puzzle about the morning star and the evening star has long been taken to be just such an example of this distinction: *the morning star = the morning star* seems to be a necessary truth, but *the morning star = the evening star* is not. Once again, these complications are the result of having introduced names or definite descriptions into the language. If we look to formulas with variables only, we have the following situation. $x = x$ is logically true, and so $\Box(x = x)$ is true. $x = y$ is not logically true; but if it is true, by (11.7) $\Box(x = y)$ is also true. Every logical truth is a necessary truth, but there are some necessary truths that are not logical truths. But then, we have a way of making the distinction our intuition wanted. While $x = y$ is, if true, necessary, just like $x = x$, it is not a logical truth. The distinction to be drawn here is not between necessary and contingent identities, but between identities that are logically—and so necessarily—true and identities that are not logically true, but are necessarily true nonetheless. Kripke (1979) made the distinction in terms of truths that are necessary and known *a priori* and truths that are necessary and known *a posteriori*.

We have seen that if x and y are the same, that fact is necessary, i.e. $(x = y) \supset \Box(x = y)$. But what if x and y are distinct? Will their distinctness also be necessary? We take the answer to be “yes.” Two objects cannot, even if circumstances change, become one. This is a different issue than whether two *names* can sometimes designate distinct objects, and sometimes the same object. Names can. Distinct objects remain distinct, no matter what.

Indiscernibility of Identicals is closely involved with the problem of *transworld identity*, one of the sticking points in the development of modal logic. In the actual world, Julius Caesar crossed the Rubicon and marched on Rome. This is, however, a contingent property of the man. It is possible that he didn't cross the Rubicon and march on Rome, and this is to say that there is another possible world in which he didn't. But how can the Julius Caesar in this world be the very same Julius Caesar as the one in the other world, when a property the Julius Caesar in this world has, the Julius Caesar in another world lacks? This problem does not depend at all upon the role of names or descriptions, but only on the object itself and its identity conditions.

But this is not a violation of The Indiscernibility of Identicals. Julius Caesar has the property of having crossed the Rubicon in this world. But there are other worlds in which he does not have the property of having crossed the Rubicon. We do not speak simply of his having a property, but only of his having a property relative to a world. We are familiar with the problem (and the solution) in the case of time. Julius Caesar was less than 4 feet tall *when he was two years old*; Julius Caesar was more than 4 feet tall *when he was eighteen years old*. We need only change the idiom to temporal worlds. Julius Caesar was less than 4 feet tall *in the temporal world in which he was two years old*; Julius Caesar was more than 4 feet tall *in the temporal world in which he was eighteen years old*. It is the same person: he had one height at age 2 and a different one at age 18.

A somewhat different problem about transworld identity is this: How can we tell which object in another world is the same as an object in this world, especially when, in another world, the object has different local properties? The problem here is not a logical one, i.e., whether an object can exist in more than one possible world having distinct characteristics in the two worlds. The problem is one of identifying an object in another world as the same object as one that is in the actual world.

This problem has some familiar intuition in the case of temporal worlds. We see a picture of grandmother's class when she was in second grade: which one in the picture is grandmother? Of course in the picture, grandmother is not an old lady but a little girl, and she has few of the visible characteristics in that picture that she has as an old lady. We can not always answer the question. Usually we need someone who remembers what she looked like as a young girl, and we rely on her ability to pick her out again after all these years. Whether or not we are ultimately successful in identifying the correct individual as grandmother, we have some understanding of the problem and some idea of how we would go about solving it.

However no such story seems appropriate for the alethic case. We do not have pictures of objects in other possible worlds, nor do we have anything like the access to them that we have for temporal worlds. Moreover, it is unlikely that just such a problem would arise in the alethic case. It is not as if we need to find out who, in another possible world, is the Julius Caesar that is in this one. Rather, the situation would seem to be whether Julius Caesar could have a certain property, i.e., whether there is a world in which he has that property. Picking a world, and trying to determine whether he has that property in that world, seems to put the cart before the horse.

Must an object retain any of its properties in every possible world? Apparently, the object must retain the property of being identical with itself. But need there be any substantial properties the object must have in every possible world? It does not seem likely.

There are two views in the philosophical literature on the nature of objects, and of the relation between objects and their properties. On one view, the *bundle theory*, an object is considered to be its collection of properties. On the other view, the *bare particular view*, the object itself is what remains after one peels away all of its

properties.² Despite initial appearances, we do not favor the bare particular theory. There is no reason to suppose that there is anything left if we peel off all of its properties. In fact, it is not clear what is meant by peeling off all of its properties. For, although Bill Clinton *is* a Democrat, in another possible world he is a Republican. This property, *possibly being a Republican* is as much a property of Bill Clinton as is the property of *actually being a Democrat*. So, as we go from world to world, we do not peel off any of his properties. As a result, it is not the bare particular view that most closely characterizes our treatment of objects in modal logic but, interestingly enough, the bundle theory.

Kripke emphasizes that possible worlds are stipulated. Typically we identify a given object, say Julius Caesar, and consider what *he* would be like in another possible world. To be sure, unless we have some substantial essential qualities for the individual, just about anything we hold true of the man is false in some possible world. But our task is not to see which object in that other world is Julius Caesar and try to identify that object via the properties it has; on the contrary, we already have the object picked out as Julius Caesar, and our task is to consider what *he* is like in another possible world. One can be too gripped by the problem of identifying him. That is not the problem. We have already identified him, and we hold him constant as we consider possible alterations.

Enough informal discussion. It is time to begin the formal semantics.

Exercises

Exercise 11.3.1 “Suppose we have a statue of Achilles that is made of bronze. The statue is melted down and reshaped into a statue of Hector. Then, at one time, this bronze equaled the statue of Achilles, but at a later time this bronze is not equal to a statue of Achilles. So there are contingent identities.” Discuss.

² We cannot resist a quote here, from the play *Peer Gynt* by Henrik Ibsen, which appeared in 1867. Peer is summarizing his history, recounting the things that have made him him, while he peels an onion and compares its layers to episodes in his life.

There lies the outermost layer, all torn; that's the shipwrecked man on the jolly boat's keel. Here's the passenger layer, scanty and thin; and yet in its taste there's a tang of Peer Gynt. Next underneath it the gold-digger ego; the juice is all gone—if it ever had any. . . . What an enormous number of swathings! Isn't the kernel soon coming to light? I'm blest if it is! To the innermost center it's nothing but swathings.

11.4 The Formal Details

From now on, one of the two-place relation symbols of our formal language from Sect. 8.1 is permanently designated to represent the equality relation. We use the familiar “=” for this purpose, and we systematically use *infix* notation, writing the standard $x = y$ instead of $=(x, y)$. We will also use the symbol informally in the usual way, but the context should make clear when we are being formal and when we are being informal.

Throughout this chapter, varying domain models or constant domain models serve equally well. In what follows, when we mention a parameter it is understood that it has no subscript if we are using constant domains, and has a subscript if we are using varying domain rules. A particular choice of $\mathbf{K}$, $\mathbf{T}$, $\mathbf{D}$, ... does not matter much either. We leave the choice open unless there is some reason to be specific. Let us say the default is varying domain $\mathbf{K}$, with other choices indicated when necessary.

We want the formal equality symbol to represent the equality relation. So, very simply, we restrict ourselves to those models in which it does so.

Definition 11.4.1 (Normal Models) A model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ is called *normal* provided, for each $\Gamma \in \mathcal{G}$ we have that $\mathcal{I}(=, \Gamma)$ is the equality relation on $\mathcal{D}(\mathcal{M})$.

That is, at each world of a normal model the equality relation symbol is interpreted to be the actual equality relation on the domain of the model. (Note that the interpretation of “=” is thus the same from world to world.) Symbolically, at a world Γ of a normal model $\mathcal{M}$,

$$\mathcal{M}, \Gamma \Vdash_v (x = y) \iff v(x) = v(y).$$

Example 11.4.2 We show the formula $(x = y) \supset \Box(x = y)$ is valid in all normal models. Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a model, with $\Gamma \in \mathcal{G}$. We want to show

$$\mathcal{M}, \Gamma \Vdash_v (x = y) \supset \Box(x = y),$$

where v is an arbitrary valuation in $\mathcal{M}$.

Well, suppose $\mathcal{M}, \Gamma \Vdash_v (x = y)$. Since $\mathcal{M}$ is a normal model, we have $v(x) = v(y)$. Now, let Δ be any member of $\mathcal{G}$ such that $\Gamma \mathcal{R} \Delta$. Since $v(x) = v(y)$, it follows that $\mathcal{M}, \Delta \Vdash_v (x = y)$. Since Δ was arbitrary, we have $\mathcal{M}, \Gamma \Vdash_v \Box(x = y)$, and this establishes the claim.

As we noted in the previous section, in a sense what makes this formula valid is the fact that in our semantics, free variables range over objects, members of possible world domains, and not over names for objects. Thus if x and y are given the same object as value, that fact persists independently of a choice of possible world. In Part V we will introduce language machinery for *naming* objects, and we will see the situation becomes much more complex then.

Incidentally, it follows easily from this example that we also have the validity of $(\forall x)(\forall y)[(x = y) \supset \Box(x = y)]$. A similar remark applies to the exercises below.

Exercises

Exercise 11.4.1 Show the validity of the following in normal modal models.

1. $(x = x)$.
2. $(x = y) \supset (y = x)$.
3. $(x = y \wedge y = z) \supset x = z$.
4. $(x = y) \supset (\Phi(x) \equiv \Phi(y))$ where $\Phi(x)$ is a formula in which y does not occur, and $\Phi(y)$ is the result of substituting occurrences of y for some, but not necessarily all, free occurrences of x in $\Phi(x)$. (Note that since y does not occur in $\Phi(x)$, y is substitutable for x since there are no quantifiers of the form $(\forall y)$.)
Hint: use Complete Induction on the complexity of Φ .

Exercise 11.4.2 Show the validity of the following in normal modal models.

1. $\Diamond(x = y) \supset (x = y)$.
2. $\neg(x = y) \supset \Box\neg(x = y)$.

11.5 Tableau Equality Rules

Tableau rules to deal with equality are simplicity itself. We build on the tableau rules for first-order modal logic, from Chap. 9, by adding rules specifically concerning equality. And for this, there are just two basic features of equality that the rules must capture. First, a thing is equal to itself, (11.1). And second, a thing is equal *only* to itself. This second principle is at the heart of “substituting equals for equals,” (11.2). If $x = y$, anything we can say about x can also be said about y because, after all, it is the same thing as saying it about x . It is the choice of earlier rules that determines which modal logic we are using, and whether it is varying or constant domain. The equality rules are essentially the same no matter what specific modal logic we work with. We state constant and varying domain rules side by side.

We begin with a rule incorporating the idea that things equal themselves. The side conditions are to make sure we don’t inadvertently introduce a new prefix/possible world onto a branch, one whose existence has not been forced on us by the modal tableau rules.

Definition 11.5.1 (Reflexivity Rules) A parameter (involving a prefix we have already seen, if varying domain) can be set equal to itself at a prefix we have already seen. Schematically we have the following.

$\overline{\sigma (p = p)}$		$\overline{\sigma (p_\tau = p_\tau)}$
Provided σ is already on the branch.	is	Provided σ and τ are already on the branch.
Constant Domain Rule		Varying Domain Rule

Note that if we are using varying domain rules, the subscript τ need not be the same as σ . Next we have a rule that covers the substitutivity of equals for equals.

Definition 11.5.2 (Atomic Substitutivity Rules) We can replace equals by equals in any *atomic* formula on the branch. Here is the constant domain version in words. For $\Phi(x)$ being an atomic formula in which x occurs free, let $\Phi(p)$ be the result of substituting occurrences of the parameter p for all free occurrences of x in $\Phi(x)$, and similarly for $\Phi(q)$. If $\sigma (p = q)$ and $\tau \Phi(p)$ both occur on a tableau branch, $\tau \Phi(q)$ can be added to the end. Similarly for varying domains. And here are the schematic versions.

For an atomic formula $\Phi(x)$:	
$\sigma (p = q)$	$\sigma (p_\rho = q_\mu)$
$\tau \Phi(p)$	$\tau \Phi(p_\rho)$
$\overline{\tau \Phi(q)}$	$\overline{\tau \Phi(q_\mu)}$
Constant Domain Rule	Varying Domain Rule

It may be wondered why substitutivity is restricted to atomic formulas. Why don't we adopt a more general substitutivity rule along the following lines.

Definition 11.5.3 (General Substitutivity Rule) We can replace equals for equals in *any* formula on a tableau branch. Schematically, exactly as in Definition 11.5.2, but without the restriction that Φ must be atomic.

For some purposes, the restricted version of Definition 11.5.2 is nicer. Since there are fewer substitutions that can be made, it may be easier to find proofs since non-atomic substitutions don't have to be taken into account. For other purposes the more general version of Definition 11.5.3 may be better since it can lead to shorter proofs because we may not need to reduce formulas to the atomic level before we substitute equals for equals, thus saving steps. Fortunately, we can show a modal tableau system allowing atomic substitutivity and the corresponding version with general substitutivity both prove the same things. We adopt the atomic version as

official, and the general version as a useful alternate, thus having the best of both.

Proposition 11.5.4 *Let $\mathbf{L}$ be any constant or varying domain first-order modal logic with equality, based on a propositional modal logic from the Lesser Modal Cube. In the prefixed tableau system for $\mathbf{L}$ replacing the Atomic Substitutivity Rule, Definition 11.5.2, by the General Substitutivity Rule, Definition 11.5.3, yields a tableau system that proves the same sentences.*

Proof Let Φ be any first-order modal sentence, allowing equality. If Φ is provable using Atomic Substitutivity, it is trivially provable using General Substitutivity because every Atomic application counts as a General application.

For the other direction, suppose we prove the following two items. First, $\mathbf{L}$ with General Substitutivity is sound and, second, $\mathbf{L}$ with Atomic Substitutivity is complete. Then by the soundness result, if Φ is provable using General Substitutivity it must be valid (in all varying or constant domain normal models as appropriate). If it is valid in all normal models, then by the completeness result it must be provable using Atomic Substitutivity.

What remains is to prove soundness when using General Substitutivity, and completeness when using Atomic Substitutivity. The first will be done in Sect. 11.6, and the second in Sects. 11.7 and 11.8.

As a matter of fact, the General Substitution rule is not the only useful rule extension that can be added to the rules above. Here are two more, and our justifications for allowing them is by showing they are *derived* rules. The first is trivial, the second a bit less so.

Branch Closure Derived Rule A constant domain tableau branch that contains $\sigma \neg(p = p)$ can be closed. Similarly a varying domain tableau branch that contains $\sigma \neg(p_\tau = p_\tau)$, can be closed. This is because in the constant domain case we can always add $\sigma (p = p)$ by the Reflexivity Rule, and then close the branch in the usual way, and similarly for the varying domain case.

Right-Left Substitutivity Derived Rule The Substitutivity Rules allow a *left-right* replacement (if we have $p = q$ with some prefix, we can replace a p occurrence with a q occurrence). But a *right-left* replacement is also possible, replacing q with p . That is, we have the following *derived* rules.

$$\begin{array}{cc}
 \sigma (p = q) & \sigma (p_\rho = q_\mu) \\
 \frac{\tau \Phi(q)}{\tau \Phi(p)} & \frac{\tau \Phi(q_\mu)}{\tau \Phi(p_\rho)} \\
 \text{Constant Domain} & \text{Varying Domain}
 \end{array}$$

We can take the constant domain version as abbreviating the following longer sequence of steps (the varying domain version is similar):

- $\sigma(p = q)$ 1.
- $\tau \Phi(q)$ 2.
- $\sigma(p = p)$ 3.
- $\sigma(q = p)$ 4.
- $\tau \Phi(p)$ 5.

In this, 1 and 2 are premises. Of course 3 is by the Reflexivity Rule. Then if we consider the formula $(x = p)$, item 3 is this with x replaced with p , and item 4 is this with x replaced with q ; consequently 4 follows from 1 and 3 by the (left-right) Substitutivity Rule. And then 5 follows from 4 and 2, by (left-right) Substitutivity again.

Example 11.5.5 Here is a varying domain **K** proof of $(\forall x)(\forall y)[(x = y) \supset \Box(x = y)]$.

- 1 $\neg(\forall x)(\forall y)[(x = y) \supset \Box(x = y)]$ 1.
- 1 $\neg(\forall y)[(p_1 = y) \supset \Box(p_1 = y)]$ 2.
- 1 $\neg[(p_1 = q_1) \supset \Box(p_1 = q_1)]$ 3.
- 1 $(p_1 = q_1)$ 4.
- 1 $\neg\Box(p_1 = q_1)$ 5.
- 1.1 $\neg(p_1 = q_1)$ 6.
- 1.1 $\neg(q_1 = q_1)$ 7.

Item 2 is from 1 by an Existential Rule, as is 3 from 2; 4 and 5 are from 3 by a Conjunctive Rule; 6 is from 5 by a Possibility Rule; 7 is from 4 and 6 by the Substitutivity Rule; now the tableau is closed using a derived rule on 7. Notice that the tableau does not close earlier because of 4 and 6. The formulas contradict each other, but the prefixes are different.

Exercises

Exercise 11.5.1 By the *closure* of a formula is meant the result of prefixing the formula with universal quantifiers for the free variables of the formula (other than parameters). For example, the closure of $[(x = y) \supset \Box(x = y)]$ is $(\forall x)(\forall y)[(x = y) \supset \Box(x = y)]$.

Give varying domain **K** tableau proofs of closures of the formulas in Exercises 11.4.1 and 11.4.2.

11.6 Tableau Soundness with Equality

In Sect. 3.3 we proved soundness of the tableau system for classical propositional logic. This was extended to modal propositional logics in Sect. 7.4, and to quantified modal logics in Sect. 9.3. The central item of each soundness proof was always the same: if a tableau branch extension rule is applied to a satisfiable tableau, the result is another satisfiable tableau. Once this has been shown, soundness follows immediately by a simple argument, and here is a sketch of it as a reminder. We show that if the sentence Φ is provable it must be valid by showing the contrapositive. Suppose Φ is not valid. Then $1 \neg \Phi$ will be satisfiable, so the construction of a proof attempt of Φ must begin with a satisfiable tableau, from which only satisfiable tableaux can result. A satisfiable tableau cannot be closed, so a proof can never be arrived at.

This argument, based on preservation of satisfiability, still applies with equality added, but we must redefine satisfiability to take equality into account. For this section we understand satisfiability as given in Definition 9.3.2 with the added condition: *models must be normal*, Definition 11.4.1. Since normal models are models, this extra condition does not invalidate our earlier satisfiability preservation results. So all that is left is to show that the equality rules of this chapter also preserve satisfiability (in normal models).

In what follows we present only the varying domain version. The constant domain version is similar. We gave two tableau rules for equality: the Reflexivity Rule and the Substitutivity Rule, and for Substitutivity we had both Atomic, Definition 11.5.2, and General versions, Definition 11.5.3. We will work with General Substitutivity, because it is what we need for the proof of Proposition 11.5.4.

Proposition 11.6.1 *A varying domain tableau with equality that is satisfiable in a normal model remains satisfiable if either the Reflexivity Rule or the General Substitutivity Rule is applied.*

Proof The proof extends those of Propositions 3.3.3 and 9.3.3 by adding two new cases, one for Reflexivity and one for General Substitutivity. We give the case for Substitutivity, and leave Reflexivity to you.

Assume we have a varying domain tableau, with equality, and in it we have a branch that is satisfiable in a normal modal model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$. Also assume that both $\sigma(p_\rho = q_\mu)$ and $\tau \Phi(p_\rho)$ occur on the branch, where $\Phi(x)$ is some arbitrary modal formula with x free. We show that if we add $\tau \Phi(q_\mu)$ to the branch, it remains satisfiable, in the same model.

Let us say the branch in question is satisfied in the normal model $\mathcal{M}$ with respect to the valuation v , using the prefix mapping θ . Then $\mathcal{M}, \theta(\tau) \Vdash_v \Phi(p_\rho)$ and $\mathcal{M}, \theta(\sigma) \Vdash_v (p_\rho = q_\mu)$. We will show that $\mathcal{M}, \theta(\tau) \Vdash_v \Phi(q_\mu)$, and this will show the extended branch is satisfied in the same model, using the same valuation.

Since $\mathcal{M}$ is normal, the interpretation of the equality symbol is by the actual equality relation, so $v(p_\rho)$ and $v(q_\mu)$ must be the same thing. Now suppose we evaluate the two formulas $\Phi(p_\rho)$ and $\Phi(q_\mu)$ at the same world of the model using

valuation v . These formulas only differ on parameters that v maps to the same thing so we should get the same results. If this is so, then since $\Phi(p_\rho)$ is true at $\theta(\tau)$ using v , the same would be the case with $\Phi(q_\mu)$, and thus the extended branch would be satisfiable using valuation v . The proof that this is in fact so is separated out as Lemma 11.6.2.

Lemma 11.6.2 *Suppose p_ρ and q_μ are parameters and $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ is a (normal) model. Then for every valuation v in $\mathcal{M}$ for which $v(p_\rho)$ and $v(q_\mu)$ are equal, and for every formula $\Phi(x)$ with only x free, and for every possible world $\Gamma \in \mathcal{G}$, $\mathcal{M}, \Gamma \Vdash_v \Phi(p_\rho)$ if and only if $\mathcal{M}, \Gamma \Vdash_v \Phi(q_\mu)$.*

Proof The proof is by Complete Induction on the complexity of $\Phi(x)$. Rather than giving all the details of the induction, we look at a representative special case. It should serve to get the ideas across. We consider the example where $\Phi(x)$ is $\Diamond(\forall y)P(x, y)$, with x and y different, so that x is free. We show the left-right implication. The other direction is, of course, similar.

For the following, $v(p_\rho) = v(q_\mu)$. Now suppose $\mathcal{M}, \Gamma \Vdash_v \Phi(p_\rho)$, that is, $\mathcal{M}, \Gamma \Vdash_v \Diamond(\forall y)P(p_\rho, y)$. Then for some $\Delta \in \mathcal{G}$ with $\Gamma \mathcal{R} \Delta$ we have $\mathcal{M}, \Delta \Vdash_v (\forall y)P(p_\rho, y)$. And then, for every y -variant w of v , $\mathcal{M}, \Delta \Vdash_w P(p_\rho, y)$. In turn, this means that $\langle w(p_\rho), w(y) \rangle \in \mathcal{I}(P, \Delta)$. Now p_ρ and q_μ are parameters, free variables that are never quantified. But y is quantified and so must be different from them. Since v and w are y variants they can only differ on y . Then $w(p_\rho) = v(p_\rho) = v(q_\mu) = w(q_\mu)$, so $\langle w(q_\mu), w(y) \rangle \in \mathcal{I}(P, \Delta)$, and thus $\mathcal{M}, \Delta \Vdash_w P(q_\mu, y)$. Since w could have been any y -variant of v , we have $\mathcal{M}, \Delta \Vdash_v (\forall y)P(q_\mu, y)$. And since $\Gamma \mathcal{R} \Delta$, $\mathcal{M}, \Gamma \Vdash_v \Diamond(\forall y)P(q_\mu, y)$, that is, $\mathcal{M}, \Gamma \Vdash_v \Phi(q_\mu)$.

11.7 Hintikka Sets with Equality

We discussed Hintikka Sets for first-order modal logic in Sect. 9.4. We worked explicitly with a varying domain version, leaving constant domain to you, and we continue that practice here when extending the earlier work to incorporate equality. Once again, **L** is one of the logics in the Lesser Modal Cube. Note the restriction to *atomic* formulas $\Phi(x)$ below. This is needed to complete the proof of Proposition 11.5.4.

Definition 11.7.1 (Hintikka Sets with Equality) A set H of prefixed formulas is a *varying domain L Hintikka set with equality* if it meets the conditions of Definition 7.5.1 (for propositional modal Hintikka sets), Definition 9.4.1 (for quantifiers), and the following (continuing the numbering from the earlier definitions).

6. H meets the following two closure conditions.

$$\sigma(p_\tau = p_\tau) \in H,$$

for every parameter p_τ and every prefix σ that occurs in H

$$\sigma(p_\rho = q_\mu) \in H \text{ and } \tau \Phi(p_\rho) \in H \implies \tau \Phi(q_\mu) \in H,$$

for every atomic formula $\Phi(x)$

The version of Hintikka set in the Definition above, with the two new conditions in item 6, is enough to get us *normal* models when we need them. A key tool in showing this is the use of *equivalence classes*, Definition 11.1.2. They are commonly used to turn an equivalence relation on a set into equality on a different set, namely the set of equivalence classes. This is exactly what happens here.

Proposition 11.7.2 (Hintikka’s Lemma) *Let H be a first-order varying domain $\mathbf{L}$ Hintikka set with equality. Then H is satisfiable in a normal varying domain $\mathbf{L}$ model.*

The proof of this is not particularly hard, but it does wander around some. Consequently we do not set something aside that we formally designate as a Proof. We simply mingle discussion and mathematics throughout the rest of this section, with the entire ensemble constituting the proof. The Lemma can be shown for any $\mathbf{L}$ from the Lesser Modal Cube, but some details vary. To keep the discussion uncluttered, for the rest of this section we assume the logic under discussion is first-order varying domain modal $\mathbf{K}$. Changes to adapt things to other logics are not difficult, and are omitted here.

Basic Assumption For everything that follows, H is a first-order varying domain $\mathbf{K}$ Hintikka set with equality.

Our goal is to show that H is satisfiable in some *normal* model. Since H meets conditions 1–5 of Definition 11.7.1, the earlier version of Hintikka’s Lemma, Proposition 9.4.3, applies and so H is satisfiable in a varying domain $\mathbf{K}$ model. The proof of Proposition 9.4.3 explicitly constructs this model using information from the Hintikka set H . We make use of the details of this construction, and we remind the reader of them now. Throughout the following we use $t, t_1, t_2, \dots$, for subscripted parameters such as p_σ . We do this when the parameter details don’t matter for what we are saying.

The model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ in which H is satisfiable, as constructed in the proof of Proposition 9.4.3, is very unlikely to be normal, but it is such that:

$\mathcal{G}$ is the set of prefixes that occur in H

$\mathcal{R}$ is such that $\sigma \mathcal{R} \sigma.n$ when both are in $\mathcal{G}$, and $\mathcal{R}$ holds in no other cases

$\mathcal{D}(\sigma)$ is non-empty and consists of all parameters p_σ with subscript σ

$\mathcal{I}(P, \sigma)$ is $\{(t_1, \dots, t_k) \mid \sigma P(t_1, \dots, t_k) \text{ is in } H\}$ for k place relation symbol P

Note that parameters play a double role: they are part of our formal language, and they also make up the domain of the model. Once this model had been specified, the proof of Proposition 9.4.3 continued as follows. We defined a valuation v_0 such that $v_0(p_\sigma) = p_\sigma$ for each parameter (that is, each parameter, as a formal symbol, names itself, as a member of the model domain). We then showed that if $\sigma \Phi \in H$ then $\mathcal{M}, \sigma \Vdash_{v_0} \Phi$, thus showing that H is satisfied in $\mathcal{M}$, using v_0 .

Our present Hintikka set H is one with equality, and so not only meets conditions 1–5 but also the new condition 6. This has as a consequence that, while the model $\mathcal{M}$ is not normal, it is nonetheless well-behaved with respect to equality. In particular, it meets the following conditions.

Proposition 11.7.3 (Equality Conditions in $\mathcal{M}$) *In the model $\mathcal{M}$, for v_0 mapping parameters to themselves:*

1. For every $\sigma \in \mathcal{G}$ and every subscripted parameter $t \in \mathcal{D}(\mathcal{M})$, $\mathcal{M}, \sigma \Vdash_{v_0} t = t$.
2. Let P be a k -place relation symbol. If $\mathcal{M}, \tau \Vdash_{v_0} P(t_1, t_2, \dots, t_k)$ and $\mathcal{M}, \sigma \Vdash_{v_0} t_1 = t'_1$, then $\mathcal{M}, \tau \Vdash_{v_0} P(t'_1, t_2, \dots, t_k)$. (And, similarly for positions 2, $\dots$, k .)
3. If $\mathcal{M}, \sigma \Vdash_{v_0} t_1 = t_2$ holds for some world σ , it holds for every world.

Proof (of Proposition 11.7.3) Item 1 is immediate from the first part of condition 6 for Hintikka sets with equality, and the fact that H is satisfiable in $\mathcal{M}$.

For item 2 recall that, in the proof of Proposition 9.4.3, atomic formulas were taken to be true in the model being constructed exactly when H explicitly said they were to be true. Then if $\mathcal{M}, \tau \Vdash_{v_0} P(t_1, t_2, \dots, t_k)$, it must be that $\tau P(t_1, t_2, \dots, t_k)$ is in H . Similarly if $\mathcal{M}, \sigma \Vdash_{v_0} t_1 = t'_1$ then $\sigma t_1 = t'_1$ is in H . And then the second part of condition 6 for Hintikka sets with equality says that $\tau P(t'_1, t_2, \dots, t_k)$ is in H , and hence $\mathcal{M}, \tau \Vdash_{v_0} P(t'_1, t_2, \dots, t_k)$.

Item 3 is a consequence of items 1 and 2 by the following argument. Suppose $\mathcal{M}, \sigma \Vdash_{v_0} t_1 = t_2$ and τ is some world other than σ . By item 1, $\mathcal{M}, \tau \Vdash_{v_0} t_1 = t_1$. Then using the substitutivity property embodied in item 2, $\mathcal{M}, \tau \Vdash_{v_0} t_1 = t_2$.

The next step is to use $\mathcal{M}$ to construct a second model which we call $\overline{\mathcal{M}}$, and a valuation $\overline{v_0}$ in it, such that $\mathcal{M}$ using v_0 and $\overline{\mathcal{M}}$ using $\overline{v_0}$ behave the same (in some sense) on each formula. But the construction will be such that $\overline{\mathcal{M}}$ is, in fact, normal. This will tell us that since H is satisfiable in $\mathcal{M}$, then H is also satisfiable in $\overline{\mathcal{M}}$, a normal model, and thus we will have established the current version of Hintikka's Lemma.

To begin, we define a relation on the domain of the model $\mathcal{M}$ as follows. For members $t_1, t_2 \in \mathcal{D}(\mathcal{M})$ (that is, for subscripted parameters), call t_1 and t_2 *equivalent* if $\mathcal{M}, \sigma \Vdash_{v_0} t_1 = t_2$ for some (any) $\sigma \in \mathcal{G}$. (We recall that $v_0(t_i) = t_i$ for each subscripted parameter.) That is, t_1 and t_2 are equivalent if the model $\mathcal{M}$ “thinks” they are equal. We denote this by $t_1 \sim t_2$.

This notion of equivalence is an equivalence relation, Definition 11.1.1. That is, it is reflexive, symmetric, and transitive. Reflexivity is part 1 of Proposition 11.7.3. We check for transitivity, and leave symmetry to you.

Suppose $t_1 \sim t_2$ and $t_2 \sim t_3$. Then there are $\sigma, \tau \in \mathcal{G}$ such that $\mathcal{M}, \sigma \Vdash_{v_0} t_1 = t_2$ and $\mathcal{M}, \tau \Vdash_{v_0} t_2 = t_3$. Using item 2 from Proposition 11.7.3 we can conclude $\mathcal{M}, \sigma \Vdash_{v_0} t_1 = t_3$, so $t_1 \sim t_3$.

For each subscripted parameter p_σ , let $\overline{p_\sigma}$ be the equivalence class containing p_σ , using the equivalence relation $\sim$. That is, $\overline{p_\sigma} = \{t \in \mathcal{D}(\mathcal{M}) \mid p_\sigma \sim t\}$. Informally we can think of $\overline{p_\sigma}$ as the set of parameters the model $\mathcal{M}$ cannot distinguish from p_σ .

Now we are ready to construct our normal model.

Normal Model Construction We form a new model $\overline{\mathcal{M}} = \langle \mathcal{G}, \mathcal{R}, \overline{\mathcal{D}}, \overline{\mathcal{I}} \rangle$ from the old model $\mathcal{M}$ as follows. The set of possible worlds and the accessibility relation are the same as in $\mathcal{M}$. The critical items are the domain and the interpretation. We define a new domain function $\overline{\mathcal{D}}$ so that equivalence classes replace members from the original model domain. Specifically, $\bar{t} \in \overline{\mathcal{D}}(\sigma)$ provided some member of the equivalence class $\bar{t}$ is in $\mathcal{D}(\sigma)$. And finally for a k -place relation symbol P , set $\langle \bar{t}_1, \dots, \bar{t}_k \rangle \in \overline{\mathcal{I}}(P, \sigma)$ provided $\langle t_1, \dots, t_k \rangle \in \mathcal{I}(P, \sigma)$.

The definition of $\overline{\mathcal{I}}$ needs comment, because it is not immediately clear that it actually is a definition. Say, for example, that P is a one-place relation symbol. Then according to the definition, $\langle \bar{t} \rangle \in \overline{\mathcal{I}}(P, \sigma)$ just in case $\langle t \rangle \in \mathcal{I}(P, \sigma)$. But it looks like we could have $\bar{t}_1 = \bar{t}_2$, while $\langle t_1 \rangle \in \mathcal{I}(P, \sigma)$ but $\langle t_2 \rangle \notin \mathcal{I}(P, \sigma)$, and if we did have this then the definition of $\overline{\mathcal{I}}$ would be ambiguous. Fortunately, it cannot happen. Suppose we have $\bar{t}_1 = \bar{t}_2$. Then $t_1 \sim t_2$, and so $\mathcal{M}, \tau \Vdash_{v_0} t_1 = t_2$, for some τ . Now if we have $\langle t_1 \rangle \in \mathcal{I}(P, \sigma)$ then we have $\mathcal{M}, \sigma \Vdash_{v_0} P(t_1)$. By part 2 of Proposition 11.7.3 we also have $\mathcal{M}, \sigma \Vdash_{v_0} P(t_2)$, and then it must be that $\langle t_2 \rangle \in \mathcal{I}(P, \sigma)$.

We thus have a well-defined model $\overline{\mathcal{M}} = \langle \mathcal{G}, \mathcal{R}, \overline{\mathcal{D}}, \overline{\mathcal{I}} \rangle$. We also define for each valuation in $\mathcal{M}$ a corresponding valuation in $\overline{\mathcal{M}}$, as follows. For a valuation v in $\mathcal{M}$, let $\bar{v}$ be the mapping from variables to the model domain of $\overline{\mathcal{M}}$ given by $\bar{v}(x) = \overline{v(x)}$.

Now we show the key facts about all this.

Model Normality The model $\overline{\mathcal{M}}$ is normal. That is, the interpretation of the equality symbol is by the equality relation on the model domain.

This is rather easy to show. By definition of $\overline{\mathcal{I}}$ we have $\langle \bar{t}_1, \bar{t}_2 \rangle \in \overline{\mathcal{I}}(=, \sigma)$ if and only if $\langle t_1, t_2 \rangle \in \mathcal{I}(=, \sigma)$, and this is if and only if $\mathcal{M}, \sigma \Vdash_{v_0} t_1 = t_2$. But by definition of $\sim$, this is equivalent to $t_1 \sim t_2$, and this in turn is equivalent to $\bar{t}_1 = \bar{t}_2$ by Proposition 11.1.3 part 3.

Model Equivalence For each formula Φ , for each $\sigma \in \mathcal{G}$, and for each valuation v in $\mathcal{M}$:

$$\mathcal{M}, \sigma \Vdash_v \Phi \iff \overline{\mathcal{M}}, \sigma \Vdash_{\bar{v}} \Phi.$$

Once Model Equivalence is established, since we know the Hintikka set H is satisfied in $\mathcal{M}$ using the valuation v_0 that maps each subscripted parameter to itself, then we also know H is satisfied in the normal model $\overline{\mathcal{M}}$ using the valuation $\overline{v_0}$.

The proof of Model Equivalence is by Complete Induction on the complexity of the formula Φ , and occupies the remainder of this section. Before we get to the induction itself we show two preliminary facts having to do with quantifier behavior, so that the flow of the proof can then proceed uninterrupted.

Valuation Fact One Suppose that v is a valuation in $\mathcal{M}$ and w is an x -variant of v at σ . Then $\overline{w}$ is an x -variant of $\overline{v}$ at σ in $\overline{\mathcal{M}}$.

Proof of Valuation Fact One Assume w is an x -variant of v at σ . That is, v and w agree on all variables except possibly x , and $w(x) \in \mathcal{D}(\sigma)$.

If y is a variable other than x , $\overline{v}(y) = \overline{v(y)} = \overline{w(y)} = \overline{w(y)}$, so $\overline{v}$ and $\overline{w}$ also agree on all variables other than x .

Since $\overline{w}(x) = \overline{w(x)}$ and $w(x) \in \overline{w(x)}$, then $w(x) \in \overline{w}(x)$. By definition, an equivalence class is in $\mathcal{D}(\sigma)$ provided some member of it is in $\mathcal{D}(\sigma)$. Since $w(x) \in \mathcal{D}(\sigma)$, then $\overline{w}(x) \in \mathcal{D}(\sigma)$, and it follows that $\overline{w}(x) \in \mathcal{D}(\sigma)$. Thus $\overline{w}$ is an x -variant of $\overline{v}$ at σ .

Valuation Fact Two Suppose that v is a valuation in $\mathcal{M}$, and hence $\overline{v}$ is a valuation in $\overline{\mathcal{M}}$. Then any x -variant of $\overline{v}$ at σ in $\overline{\mathcal{M}}$ must be of the form $\overline{w}$ where w is a valuation in $\mathcal{M}$ that is an x -variant of v at σ .

Proof of Valuation Fact Two Assume v is a valuation in $\mathcal{M}$, and so $\overline{v}$ is a valuation in $\overline{\mathcal{M}}$. Also assume that $\overline{u}$ is an x -variant of $\overline{v}$ at σ in $\overline{\mathcal{M}}$. We now define a valuation w in $\mathcal{M}$, and show that it is an x variant of v at σ and $\overline{u} = \overline{w}$.

Valuation w in $\mathcal{M}$ is defined as follows. On variables z other than x , set $w(z) = v(z)$. Of course we still must define $w(x)$. Since $\overline{u}$ is an x -variant of $\overline{v}$ at σ , $\overline{u}(x) \in \mathcal{D}(\sigma)$. By definition of $\mathcal{D}$, for this to happen some member of the equivalence class $\overline{u}(x)$ must be in $\mathcal{D}(\sigma)$. Choose one such member and set $w(x)$ to be it. This completes the definition of a valuation w in $\mathcal{M}$. We next show it has the properties we want.

By its definition, $w(x) \in \overline{u}(x) = \overline{u(x)}$. Then the equivalence classes $\overline{w(x)}$ and $\overline{u(x)}$ have the member $w(x)$ in common, so $\overline{w(x)} = \overline{u(x)}$ by Proposition 11.1.3 part 2, and so $\overline{w}(x) = \overline{u}(x)$. Also, if z is a variable other than x , $w(z) = v(z) \in \overline{v}(z) = \overline{u}(z)$ (since $\overline{u}$ and $\overline{v}$ differ only on x). Then for $z \neq x$, $\overline{w}(z) = \overline{u}(z)$, since these equivalence classes have $w(z)$ in common. It follows that for the valuation w we constructed, $\overline{w} = \overline{u}$. And further, by construction w is an x -variant of v at σ .

Now we turn to the proof of Model Equivalence itself, that we always have the following.

$$\mathcal{M}, \sigma \Vdash_v \Phi \iff \overline{\mathcal{M}}, \sigma \Vdash_{\overline{v}} \Phi \quad (11.8)$$

As we noted earlier, this fundamental connection between the models $\mathcal{M}$ and $\overline{\mathcal{M}}$ is shown by complete induction on the complexity of Φ .

Assume as an *induction hypothesis* that (11.8) holds for all valuations v , all worlds σ , and all formulas Ψ that are less complex than Φ . We show that, under this hypothesis, (11.8) also holds for Φ . The argument has several cases, depending on the form of Φ .

Atomic Case Say Φ is $P(z_1, \dots, z_k)$. This case does not actually use the induction hypothesis.

$$\begin{aligned}
 \mathcal{M}, \sigma \Vdash_v P(z_1, \dots, z_k) &\iff \langle v(z_1), \dots, v(z_k) \rangle \in \mathcal{J}(P, \sigma) && \text{atomic truth def} \\
 &\iff \langle \overline{v(z_1)}, \dots, \overline{v(z_k)} \rangle \in \overline{\mathcal{J}}(P, \sigma) && \overline{\mathcal{J}} \text{ def} \\
 &\iff \langle \overline{v}(z_1), \dots, \overline{v}(z_k) \rangle \in \overline{\mathcal{J}}(P, \sigma) && \overline{v} \text{ def} \\
 &\iff \overline{\mathcal{M}}, \sigma \Vdash_{\overline{v}} P(z_1, \dots, z_k) && \text{atomic truth def}
 \end{aligned}$$

Propositional Cases If Φ is a negation, conjunction, disjunction, or implication, the argument is straightforward and is left to you.

Modal Cases Say Φ is $\Box\Psi$ and by the induction hypothesis (11.8) holds for Ψ . This is used in the transition between the first two lines below.

$$\begin{aligned}
 \mathcal{M}, \sigma \Vdash_v \Box\Psi &\iff \mathcal{M}, \tau \Vdash_v \Psi && \text{for all } \tau \text{ with } \sigma \mathcal{R} \tau \\
 &\iff \overline{\mathcal{M}}, \tau \Vdash_{\overline{v}} \Psi && \text{for all } \tau \text{ with } \sigma \mathcal{R} \tau \\
 &\iff \overline{\mathcal{M}}, \sigma \Vdash_{\overline{v}} \Box\Psi
 \end{aligned}$$

The case where Φ is $\Diamond\Psi$ is similar and is omitted.

Quantifier Cases We only consider the existential case since the other is similar; say Φ is $(\exists x)\Psi$, where (11.8) holds for Ψ . We give the two directions of the argument separately.

Suppose first that we have $\mathcal{M}, \sigma \Vdash_v (\exists x)\Psi$. Then we have $\mathcal{M}, \sigma \Vdash_w \Psi$ where w is some x -variant of v at σ . By the induction hypothesis this implies $\overline{\mathcal{M}}, \sigma \Vdash_{\overline{w}} \Psi$, and by *Valuation Fact One*, $\overline{w}$ is an x -variant of $\overline{v}$ at σ . Then $\overline{\mathcal{M}}, \sigma \Vdash_{\overline{v}} (\exists x)\Psi$. For the other direction, suppose we have $\overline{\mathcal{M}}, \sigma \Vdash_{\overline{v}} (\exists x)\Psi$. Then there is a valuation $\overline{u}$, in $\overline{\mathcal{M}}$, that is an x -variant of $\overline{v}$ at σ , such that $\overline{\mathcal{M}}, \sigma \Vdash_{\overline{u}} \Psi$. By *Valuation Fact Two*, there is an x -variant w of v in $\mathcal{M}$ such that $\overline{w} = \overline{u}$. But then $\overline{\mathcal{M}}, \sigma \Vdash_{\overline{w}} \Psi$. By the induction hypothesis, $\mathcal{M}, \sigma \Vdash_w \Psi$. And since w is an x -variant of v at σ , $\mathcal{M}, \sigma \Vdash_v (\exists x)\Psi$.

This completes the inductive argument, establishes (11.8), and completes the proof for Hintikka's Lemma, Proposition 11.7.2.

11.8 Tableau Completeness with Equality

Completeness for modal tableaux with equality is rather easy to prove, because much of the hard work was done in Chap. 9. As in earlier chapters we can make use of a systematic tableau construction, or use maximal consistency. We only sketch the systematic construction version, and we only do this for first-order **K** with varying domains, and equality. Adapting things to constant domains, or to other logics from the Lesser Modal Cube is not difficult, and is left to those who want a technical workout. The same applies to the maximal consistency approach.

As usual, completeness is shown in the contrapositive direction. That is, one shows that if a sentence Φ is not provable using the tableau rules, including those for equality from this Chapter, then Φ can be falsified in some normal model. And as noted, we only consider the varying domain **K** rules. The argument is essentially the same as the one in Chap. 9. As part of that, we gave a systematic tableau construction procedure for first-order varying domain **K** tableaux, and we build on that. Recall that $\rho_1, \rho_2, \rho_3, \dots$ is a listing of all subscripted parameters. We note that the construction is explicitly carried through for a particular example in Sect. 11.9, and you may want to look at that if you find the presentation here heavy going.

Systematic Tableau Construction Additions In Sect. 9.5 we gave a systematic tableau construction procedure for varying domain **K**. We now modify that to take the equality rules into account, and this is quite a simple business. The construction from Sect. 9.5 had seven clauses. We now insert the following after clause 7. All the rest remains the same. The new conditions are as follows.

8. For each prefix σ that has been introduced to the branch, add $\sigma p_\tau = p_\tau$ to the branch end where: τ occurs as a prefix on the branch and p_τ is the first item in our enumeration $\rho_1, \rho_2, \rho_3, \dots$ of subscripted parameters such that $\sigma p_\tau = p_\tau$ does not already occur on the branch.
9. For each $\sigma p_\rho = q_\mu$ and for each atomic $\tau \Phi(p_\rho)$ that occur on the branch, add $\tau \Phi(q_\mu)$ to the branch end, provided this prefixed formula does not already occur on the branch.

These complete the additions to the systematic tableau construction procedure.

Proposition 11.8.1 (Tableau Completeness with Equality) *If the sentence Φ is valid in all normal varying domain first-order **K** models, Φ has a tableau proof using the varying domain **K** rules and the equality rules.*

Proof Suppose Φ has no tableau proof. Apply the systematic tableau construction procedure, including the additions to handle the equality rules. It cannot produce a closed tableau, so an open branch $\mathcal{B}$ is generated, probably infinite. The set of prefixed formulas on this open branch is easily seen to be a Hintikka Set With Equality, Definition 11.7.1. Call this set H . By Hintikka's Lemma, Proposition 11.7.2, H is satisfiable in a normal varying domain **K** model, call it $\mathcal{M}$. Since the tableau begins

with $1 \rightarrow \Phi$ then this prefixed formula must be in H , and hence is satisfiable in $\mathcal{M}$. Then Φ is not valid in all normal models.

We have dealt with only a single case: varying domain **K**. But the other cases are just variations on this one. We stop here.

11.9 An Example

The completeness proof for tableaux with equality is full of details, mostly concentrated in the proof of Hintikka's Lemma and its creation of a normal model. It is easy to lose the big picture. In this section we illustrate the general ideas through an example. But first, we must deal with an expository problem.

Condition 5 in the definition of Hintikka set, Definition 9.4.1, generally forces Hintikka sets to be infinite: if a Hintikka set contains a prefixed formula with a universal quantifier, or a negated existential quantifier, *every* parameter instance must be present. On top of this, condition 8 of the Systematic Tableau Construction Additions also forces the Hintikka sets we construct to be infinite. Setting things up as we did makes certain things in the completeness proof easier to state and work with. But there are somewhat more complex conditions that could have been used and that allow Hintikka sets to be finite sometimes. For instance, we could have required that if $\sigma(\forall x)\Psi(x)$ is present in a Hintikka set then: for at least one parameter p_σ , $\sigma\Psi(p_\sigma)$ is present; and for every parameter q_σ that occurs in the Hintikka set, $\sigma\Psi(q_\sigma)$ is present. Similarly for negated existentials, and for reflexivity conditions on equality. We just mention this version, and do not prove that it works. Actually, going through a proof of Hintikka's Lemma with this alternative condition would be a good exercise. Anyway, we use the alternative version for this section so that we can fully present a finite example of interest. Specifically, we attempt to give a varying domain **K** proof of

$$[(\forall x)\Box(\exists y)(x = y) \wedge (\exists x)P(x)] \supset [\Diamond(\exists x)P(x) \supset (\exists x)\Diamond P(x)]$$

and from our failure, we construct a *normal* model falsifying it. A failed tableau is given in Fig. 11.1. It is not constructed strictly using our systematic procedure. Still, we have done enough to allow the extraction of a counter-model from it. We can tell we have done enough because the set of formulas on the single open branch is a Hintikka set with equality, but using the alternative version of Hintikka set described above.

We first use this set of formulas on the open branch, the Hintikka set, to construct a non-normal model, $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$, by doing just what the Hintikka set “tells” us to do. This is as described in the proof of Proposition 9.4.3. The possible worlds are the prefixes, so $\mathcal{G} = \{1, 1.1\}$, and we have $1\mathcal{R}1.1$. The domain of model $\mathcal{M}$ is $\{p_1, p_{1.1}, q_{1.1}\}$, where $\mathcal{D}(1) = \{p_1\}$ and $\mathcal{D}(1.1) = \{p_{1.1}, q_{1.1}\}$. As to the

- 1 $\neg\{[(\forall x)\Box(\exists y)(x=y) \wedge (\exists x)P(x)] \supset$
- $[\Diamond(\exists x)P(x) \supset (\exists x)\Diamond P(x)]\}$ 1.
- 1 $(\forall x)\Box(\exists y)(x=y) \wedge (\exists x)P(x)$ 2.
- 1 $\neg[\Diamond(\exists x)P(x) \supset (\exists x)\Diamond P(x)]$ 3.
- 1 $(\forall x)\Box(\exists y)(x=y)$ 4.
- 1 $(\exists x)P(x)$ 5.
- 1 $\Diamond(\exists x)P(x)$ 6.
- 1 $\neg(\exists x)\Diamond P(x)$ 7.
- 1 $P(p_1)$ 8.
- 1 $\Box(\exists y)(p_1=y)$ 9.
- 1 $\neg\Diamond P(p_1)$ 10.
- 1.1 $(\exists x)P(x)$ 11.
- 1.1 $(\exists y)(p_1=y)$ 12.
- 1.1 $\neg P(p_1)$ 13.
- 1.1 $P(p_{1.1})$ 14.
- 1.1 $p_1 = q_{1.1}$ 15.
- 1 $P(q_{1.1})$ 16.
- 1.1 $\neg P(q_{1.1})$ 17.
- 1 $p_1 = p_1$ 18.
- 1 $p_{1.1} = p_{1.1}$ 19.
- 1 $q_{1.1} = q_{1.1}$ 20.
- 1 $p_1 = q_{1.1}$ 21.
- 1 $q_{1.1} = p_1$ 22.
- 1.1 $p_1 = p_1$ 23.
- 1.1 $p_{1.1} = p_{1.1}$ 24.
- 1.1 $q_{1.1} = q_{1.1}$ 25.
- 1.1 $q_{1.1} = p_1$ 26.

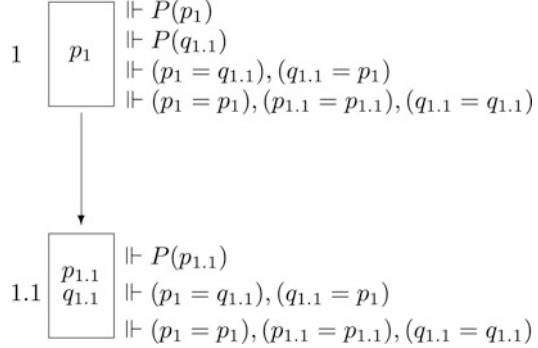
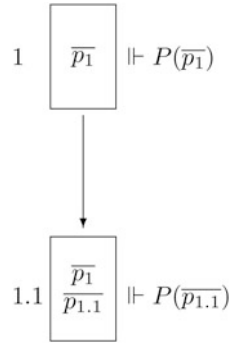
Items 2 and 3 are from 1 by a Conjunctive Rule, as are 4 and 5 from 2, and 6 and 7 from 3; 8 is from 5 by an Existential Rule; 9 is from 4 by a Universal Rule, as is 10 from 7; 11 is from 6 by a Possibility Rule; 12 is from 9 by a Necessity Rule, as is 13 from 10; 14 is from 11 by an Existential Rule, as is 15 from 12; 16 is from 8 and 15 by the Substitutivity Rule, as is 17 from 13 and 15; 18, 19, and 20 are by the Reflexivity Rule; 21 is from 15 and 18 by Substitutivity, as is 22; 23, 24, and 25 are by Reflexivity, and 26 is from 15 and 23 by Substitutivity. At this point we have an unclosed tableau, and the set of prefixed formulas on the only open branch is a Hintikka set in the loose sense described above..

Fig. 11.1 A failed tableau using equality

interpretation, formula 16 of the tableau is 1 $P(q_{1.1})$, so at world 1 we interpret P to hold of $q_{1.1}$. And so on.

We leave it to you to check that the Hintikka set formed from the open branch of the tableau in Fig. 11.1 yields the model $\mathcal{M}$ shown schematically in Fig. 11.2. We also leave it to you to verify that the Hintikka set is, in fact, satisfied in this model. Note that the model is *not* normal, because at world 1.1 we have $p_1 = q_{1.1}$ (because of item 15 of the tableau) though p_1 and $q_{1.1}$ are different objects.

We next define a second model, $\overline{\mathcal{M}} = \langle \mathcal{G}, \mathcal{R}, \overline{\mathcal{D}}, \overline{\mathcal{I}} \rangle$, based on the first one as described in the proof of Proposition 11.7.2. It is shown schematically in Fig. 11.3. Here are the construction details. We begin by defining a relation $\sim$ on the domain of the model $\{p_1, p_{1.1}, q_{1.1}\}$ by setting $x \sim y$ to hold if some world of model $\mathcal{M}$ “thinks” $x = y$. So we have $p_1 \sim p_1$, $p_{1.1} \sim p_{1.1}$, $q_{1.1} \sim q_{1.1}$, $p_1 \sim q_{1.1}$, and

Fig. 11.2 A non-normal counter model $\mathcal{M}$ **Fig. 11.3** A normal counter model $\overline{\mathcal{M}}$ 

$q_{1.1} \sim p_1$. Then we form equivalence classes, where $\overline{x} = \{y \mid y \sim x\}$. This gives us just two distinct classes.

$$\begin{aligned}\overline{p_1} &= \overline{q_{1.1}} = \{p_1, q_{1.1}\} \\ \overline{p_{1.1}} &= \{p_{1.1}\}\end{aligned}$$

The worlds and the accessibility relation of $\overline{\mathcal{M}}$ are the same as in $\mathcal{M}$. The domain of the model $\overline{\mathcal{M}}$ is the set of equivalence classes, $\{\overline{p_1}, \overline{p_{1.1}}\}$. We say an equivalence class is in the domain of a particular world in $\overline{\mathcal{M}}$ if some member of it is in the domain of that world in the model $\mathcal{M}$. Then $\overline{p_1}$ is in the domain of world 1 in $\overline{\mathcal{M}}$ since p_1 is a member of $\overline{p_1}$, and is in the domain of world 1 in $\mathcal{M}$. Likewise $\overline{p_1}$ is in the domain of world 1.1 in $\overline{\mathcal{M}}$ since $q_{1.1}$ is a member, and $q_{1.1}$ is in the domain of world 1.1 in the model $\mathcal{M}$. Similarly $\overline{p_{1.1}}$ turns out to be in the domain of world 1.1 in $\overline{\mathcal{M}}$, but it is not in the domain of world 1.

We interpret P in the new model $\overline{\mathcal{M}}$ by saying it holds of an equivalence class, at a world, if it holds of its members, at that world in the model $\mathcal{M}$. Thus, for instance, P holds of $\overline{p_1}$ at world 1 in $\overline{\mathcal{M}}$ since $\overline{p_1} = \{p_1, q_{1.1}\}$ and P holds of both p_1 and $q_{1.1}$ at world 1 in the model $\mathcal{M}$.

The interpretation of the relation symbol $=$ is equality, so we do not indicate it in the diagram of Fig. 11.3.

In the model $\overline{\mathcal{M}}$, $\overline{p_1}$ is common to both worlds. This is essential for the formula $(\forall x)\Box(\exists y)(x = y)$ to be true at world 1. But we leave it to you to verify that this new normal model $\overline{\mathcal{M}}$ does indeed falsify the formula we failed to prove.

Exercises

Exercise 11.9.1 Give a varying domain **K** proof, using the equality rules, of the following sentence.

$$(\forall x)\Box(\exists y)(x = y) \supset [(\exists x)\Diamond P(x) \supset \Diamond(\exists x)P(x)]$$

Exercise 11.9.2 Construct a normal, varying domain, **K** model in which the following sentence is falsified.

$$(\forall x)\Box(\exists y)(x = y) \supset [(\forall x)\Box P(x) \supset \Box(\forall x)P(x)]$$

Do so by attempting a tableau proof as we did above, and extracting a counter-model from it.

References

- Carnap, R. (1947). *Meaning and necessity: A study in semantics and modal logic*. Chicago: University of Chicago Press.
- Frege, G. (1879). *Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens*. Partial translation in Frege (1952). Halle: Verlag von L. Nebert.
- Frege, G. (1892). Über Sinn und Bedeutung. *Zeitschrift für Philosophie und philosophische Kritik*, 100, 25–50. “On Sense and Reference” translated in Frege (1952).
- Frege, G. (1952). In P. Geach & M. Black (Eds.), *Translations from the philosophical writings of Gottlob Frege*. Oxford: Basil Blackwell.
- Kripke, S. (1979). A puzzle about belief. In A. Margalit (Ed.), *Meaning and use* (pp. 239–283). Dordrecht: D. Reidel.
- Mendelsohn, R. L. (1982). Frege’s Begriffsschrift theory of identity. *Journal of the History of Philosophy*, 20, 279–299.
- Russell, B. (1905). On denoting. *Mind*, 14, 479–493. Reprinted in Robert C. Marsh (Ed.), *Logic and knowledge: Essays 1901–1950, by Bertrand Russell*. London: Allen & Unwin (1956).
- Smullyan, A. F. (1948). Modality and description. *The Journal of Symbolic Logic*, 13, 31–37.
- Wittgenstein, L. (1922). *Tractatus Logico-Philosophicus*. tr. by D. F. Pears & B. F. McGuinness, Routledge & Kegan Paul, London, 1961. *Annalen der Naturphilosophie*.

Chapter 12

Existence



In Chap. 8 we saw that two basic kinds of quantification were natural in first-order modal logics: possibilist and actualist. Possibilist quantifiers range over what *might* exist. This corresponds semantically to constant domain models where the common domain is, intuitively, the set of things that could exist. We also saw that in the possibilist approach we could introduce an existence primitive (Sect. 8.9) and relativize quantifiers to it, permitting the possibilist approach to paraphrase the actualist version.

Now it is time to consider things from the other side. In this chapter we take the actualist quantifier as basic. Our models are varying domain, and we think of the domain of each world as what actually exists at that world. At each world quantifiers range over the domain of that world only. We will see that this allows us to *define* an existence formula, rather than taking one as primitive. Further, we will be able to formulate single-formula versions of the Barcan and the Converse Barcan schemes. And this, in turn, will allow us to use varying domain machinery to reason in constant domain settings. In other words, the actualist quantifier allows us to talk in a possibilist mode when we want to. Incidentally, we will see that what are called monotonic and anti-monotonic semantics can be captured by this approach as well.

The upshot of this and our earlier work in Chap. 8 is that a choice between the possibilist and the actualist quantifier cannot be decided on formal grounds. With good will on both sides, a possibilist philosopher and an actualist philosopher can talk meaningfully to each other—it probably happens all the time. It will take a bit more work to get each to speak meaningfully to a classical philosopher. The hurdle in this case is that the modal logicians believe it is meaningful to speak of things that do not exist, whereas classically, this has been thought to be incoherent. After presenting some of the formal details, we will rehearse the issues and arguments surrounding this difficult issue.

12.1 To Be

The basic semantics of this chapter is varying domain—skeletons have a domain function that assigns to each possible world a domain of quantification for that world, and different worlds are allowed to have different domains. (We also assume all models are normal—the equality symbol is interpreted by the equality relation on the domain of the skeleton. We will not say this each time.) The value that a valuation assigns to a free variable, x , in a varying domain model must be something in the domain of the *model*. That is, it must be something in the domain of *some* world, but at any given world that value may or may not exist. If the value assigned to x exists at world Γ , that is, if it is in the domain of Γ , quantifiers at Γ have that value in their range, and if the value assigned to x does not exist at Γ , that value is out of quantifier range. This means we can make use of the well-known formula $(\exists y)(y = x)$ to express the existence of the value of x at a world. The following Proposition makes this idea precise; the proof is simple, and is left to you.

Proposition 12.1.1 *Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a normal, varying domain model, and let v be a valuation. Then $\mathcal{M}, \Gamma \Vdash_v (\exists y)(y = x)$ if and only if $v(x) \in \mathcal{D}(\Gamma)$, that is, if and only if $v(x)$ exists at world Γ .*

This says we can use $(\exists y)(y = x)$ in a varying domain model to represent the existence of x within our formal language. The formula plays a fundamental role, so we introduce special notation for it.

Definition 12.1.2 (Existence Formula) For a variable x , $E(x)$ abbreviates $(\exists y)(y = x)$, where y is a variable distinct from x .

$E(x)$ is true at a world of a varying domain model with respect to a valuation if and only if the value assigned to x exists at that world. Now that we can explicitly assert existence, certain peculiarities of varying domain semantics can be dealt with naturally and easily. For instance, the classically valid formula $(\forall x)\Phi(x) \supset \Phi(z)$ is not valid in varying domain modal semantics. At a world Γ of a varying domain model, the value assigned to the free variable z in $\Phi(z)$ might not exist, and so not be in the range of the quantifier $(\forall x)$. Then, even though everything that exists at Γ makes $\Phi(x)$ true, the value assigned to z need not. But if we knew the value assigned to z did, in fact, exist at Γ , things would be different. The formula $[(\forall x)\Phi(x) \wedge E(z)] \supset \Phi(z)$ is valid. We leave it to you to check this. Thus we retain universal instantiation for those variables that actually designate existent objects. This may be unfamiliar because in classical logic variables are *assumed* to designate existent objects, so $E(z)$ is always satisfied and consequently is omitted.

Please note, $E(x)$ is a defined formula, to be contrasted with $\mathcal{E}(x)$ from Sect. 8.9 which makes use of a special predicate symbol, set aside for the purpose of representing equality. The formula version $E(x)$ will only be used when varying domain semantics is under consideration. In a constant domain model, $E(x)$ is simply true for all values of x . That is, $(\forall x)E(x)$ is valid in constant domain models, and thus is not particularly interesting. The special predicate $\mathcal{E}$ can be interpreted

differently at different possible worlds, and thus is of use to us in working with constant domain models. On the other hand, we don't need a special predicate like $\mathcal{E}$ in varying domain models because we have a definable formula $E(x)$ that we can use. $\mathcal{E}$ has no particular interest for us in varying domain models.

There is a family of logics called *free logics*, one of whose characteristic features is that variables need not designate. This was discussed in Sect. 8.8. The formal machinery for varying domain modal logic, in fact, gives us a free logic. (There are more things in heaven and earth, Horatio, than are dreamt of by your quantifiers.)

Exercises

Exercise 12.1.1 Prove Proposition 12.1.1.

12.2 Tableau Proofs

Varying domain quantifier rules were given in Sect. 9.2, and rules for equality were given in Chap. 11. Since quantification and equality is all that is involved in $E(z)$, we need no new tableau machinery to deal with formulas that contain it.

Example 12.2.1 Here is a varying domain proof of $(\forall z)E(z)$. Since quantifiers intuitively range over what exists, this states the obvious fact that everything that exists, exists. As it happens, modal issues are not involved in this example.

$$\begin{aligned} 1 \neg(\forall z)E(z) & \quad 1. \\ 1 \neg(\forall z)(\exists y)(y = z) & \quad 2. \\ 1 \neg(\exists y)(y = p_1) & \quad 3. \\ 1 \neg(p_1 = p_1) & \quad 4. \end{aligned}$$

In this, 2 is 1 unabbreviated; 3 is from 2 by an Existential rule; 4 is from 3 by a Universal rule. Now closure is by a (derived) equality rule.

There are a couple of derived tableau rules that often simplify the use of $E(x)$ in tableau construction. The varying domain universal quantification rule only permits the use of parameters having the same subscript as the prefix—that is, from $\sigma (\forall x)\Phi(x)$ we can get $\sigma \Phi(p_\sigma)$, but not $\sigma \Phi(p_\tau)$ where $\tau \neq \sigma$. However, if we “know” p_τ exists at σ , we should be able to conclude $\sigma \Phi(p_\tau)$ after all. The following makes this precise.

Definition 12.2.2 (Derived Parameter Existence Rule) A varying domain tableau branch containing $\sigma E(p_\tau)$ and $\sigma (\forall x)\Phi(x)$ can be extended with $\sigma \Phi(p_\tau)$. There

is a similar rule for branches containing $\sigma \neg(\exists x)\Phi(x)$. Schematically, we have the following.

$$\frac{\sigma (\forall x)\Phi(x)}{\sigma \mathbf{E}(p_\tau)} \quad \frac{\sigma \neg(\exists x)\Phi(x)}{\sigma \mathbf{E}(p_\tau)} \\ \sigma \Phi(p_\tau) \quad \sigma \neg\Phi(p_\tau)$$

Here is the justification for this derived rule. Suppose, on a tableau branch, we have $\sigma \mathbf{E}(p_\tau)$ and also $\sigma (\forall x)\Phi(x)$. Then $\sigma \Phi(p_\tau)$ can be added, by the following sequence of legitimate moves.

- $\sigma \mathbf{E}(p_\tau)$ 1.
- $\sigma (\forall x)\Phi(x)$ 2.
- $\sigma (\exists y)(y = p_\tau)$ 3.
- $\sigma q_\sigma = p_\tau$ 4.
- $\sigma \Phi(q_\sigma)$ 5.
- $\sigma \Phi(p_\tau)$ 6.

Item 3 is item 1 unabbreviated; 4 is from 3 by an existential rule where q_σ is a new parameter; 5 is from 2 by a universal rule; 6 is from 4 and 5 by Substitutivity.

Here is another useful derived tableau rule. We leave its proof as an exercise. The intuition is that, while free variables might have nonexistent objects as values, the rules for parameters in varying domain tableau proofs ensure they always designate existents.

Definition 12.2.3 (Parameter NonExistence Rule) A branch of a varying domain tableau containing $\sigma \neg\mathbf{E}(p_\sigma)$ closes.

Exercises

Exercise 12.2.1 Justify the Parameter NonExistence derived rule.

Exercise 12.2.2 Show that the formula $[(\forall x)\Phi(x) \wedge \mathbf{E}(z)] \supset \Phi(z)$ is valid in all varying domain models, where z is substitutable for x in $\Phi(x)$, that is, in $\Phi(x)$, no free occurrence of x is within the scope of a quantifier whose variable is z .

Exercise 12.2.3 Give a varying domain **K** proof of the following sentence.

$$[(\forall x)\Diamond\mathbf{E}(x) \wedge (\exists x)\Box P(x)] \supset \Diamond(\exists x)P(x)$$

Exercise 12.2.4 Attempt to give a varying domain **K** proof of the following sentence.

$$(\exists x)\Box P(x) \supset \Diamond(\exists x)P(x)$$

From the failed attempt, construct a counter-model.

12.3 The Paradox of NonBeing

We have been speaking about things that do not exist. Eventually we will show that the existence of unicorns, for example, can be formally denied in our models using definite descriptions and an existence predicate. But over the centuries philosophers have found speaking of nonexistence to be very problematic. In this and the next few sections we discuss some of the issues that are involved.

Puzzles and conundrums abound if one tries either to assert or to deny that something exists. The oldest among them finds its inspiration in some fragments of Parmenides, who cautioned against saying “what is not”:

Come now and I will tell thee—listen and lay my word to heart—the only ways of inquiry that are to be thought of: one, that [That which is] is, and it is impossible for it not to be, is the Way of Persuasion, for Persuasion attends on Truth.

Another, that It is not, and must needs not be—this, I tell thee is a path that is utterly undiscernible; for thou couldst not know that which is not—for that is impossible—nor utter it.

For it is the same thing that can be thought and that can be.

What can be spoken of and thought must be; for it is possible for it to be, but it is not possible for “nothing” to be. These things I bid thee ponder; for this is the first way of inquiry from which I hold thee back. (Cornford 1957, fragments 2,3,6 ll.1-3)

The ancient Greeks used “is” to express predication, to express identity, to express existence, and even to express truth; “is not,” accordingly, could mean that a thing lacked a (or any) property, that it was not identical (with something), that it failed to exist, or that it was false. The ambiguity of his injunction rendered Parmenides’s remarks exceedingly rich and evocative: a goldmine for philosophical excavation. The great craftsman Plato could not resist applying his logical spade to this archaeological dig, painstakingly separating out layers of different meanings in the sediment that the pressures of thought and speech had forced together. His results he recorded as a dialogue between Socrates and the mysterious Eleatic Stranger in the *Sophist*.

In that dialogue, Plato distinguished the “is” of predication from the “is” of identity, and even, some commentators would add, the “is” of existence. He determined that Parmenides was wrong if taken to urge that one can neither assert nor believe what is not, in the sense that one can neither assert nor believe false propositions. But he found an impassable kernel in Parmenides’s injunction: speaking and thinking about nonexistent objects. Quine characterized the predicament in which he landed *Plato’s Beard*: “historically,” Quine says, “it has proved tough, frequently dulling the edge of Occam’s Razor,” (Quine 1948, p. 2). Here are some highlights from Plato’s discussion.

The Eleatic Stranger says “that the term ‘what is not’ must not be applied to anything that exists; ... and since it cannot be applied to what exists, neither can it properly be applied to ‘something.’ ” From which he concludes that “to speak of what is not ‘something’ is to speak of no thing at all.” He echoes Parmenides:

Must we not even refuse to allow that in such a case a person is *saying* something, though he may be speaking of nothing? Must we not assert that he is not even saying anything when he sets about uttering the sounds 'a thing that is not'?

Matters get worse. "[T]he nonexistent reduces even one who is refuting its claims to such straits that, as soon as he sets about doing so, he is forced to contradict himself," continues the Stranger. For when he spoke of the nonexistent as "*being* a thing not to be uttered or spoken of or expressed," he applied the term 'being' to it; moreover, "in speaking of it as 'a thing not to be expressed or spoken of or uttered,' [he] was . . . using language as if referring to a single thing." And so he ends plaintively:

In that case there is nothing to be said for me. I shall be found to have had the worst of it, now and all along, in my criticism of the nonexistent.

Parmenides, then, appears vindicated: we are reduced to babbling or incoherence or self-contradiction if we try to speak about the utterly nonreal.

Thus we come to a genuine philosophical paradox, *The Paradox of NonBeing*: to say of a thing that is not that it is not is to say that it is something, viz., a thing that is not, and so it is. Here is a modern formulation due to (Cartwright 1960, 21):

To deny the existence of something—of unicorns, for example—we must indicate what it is the existence of which is being denied; and this requires that unicorns be referred to or mentioned; the negative existential must be about them. But things which do not exist cannot be referred to or mentioned; no statement can be about them. So, given that we have denied their existence, unicorns must after all exist. The apparently true negative existential is thus either false or not really a statement at all; and, since the argument applies as well in any other case, we seem forced to conclude that there are no true negative existentials.

The conclusion of the argument is unacceptable. But the argument is valid. From the two premises,

- (A) If an individual (call him "John") denies the existence of something, then John refers to what he says does not exist;
- (B) Things which do not exist cannot be referred to or mentioned; no statement can be about them.

the following conclusion is drawn:

- (C) If John denies the existence of something, then what John says does not exist does exist.

This argument is logically straightforward. It requires little more than transitivity of implication. We need only to rephrase premise (B) to read

- (D) If John refers to or mentions something, Ks, then Ks exist.

Then, as an instance of (D), taking "Ks" to be "what John says does not exist," we have

- (E) If John refers to or mentions what he says does not exist, then what John says does not exist does exist.

And, from (A) and (E), by transitivity, we reach the desired conclusion (C).

The argument can be generalized. By parallel reasoning, we can show that one cannot think or believe anything about something that does not exist. And this is an equally unpalatable conclusion. Responses to the paradox have sorted themselves into two groups. Berlin (1949–1950) called them *Inflationists* and *Deflationists*. Inflationists accept premise (A) and deny premise (B); Deflationists accept premise (B) and deny premise (A).

12.4 Deflationists

The Deflationists were the dominant group in twentieth century Anglo-American philosophy. They accept premise (B) of Sect. 12.3 but reject premise (A): existence is a necessary condition for reference, but no such semantic relation comes into play when denials of existence are made.

The main support for the Deflationist view is found in Russell (1905). We shall discuss Russell's analysis of names and definite descriptions in Chap. 20. But the flavor of his solution to the Paradox of NonBeing can be given here.

Before Russell, one would have thought that a sentence like

$$\text{Winged horses exist} \tag{12.1}$$

would, be treated as a subject/predicate sentence, just like

$$\text{Winged horses fly.} \tag{12.2}$$

(12.1) would be analyzed in such a way that the term “Winged horses” refers to some things and the sentence says about these things that they have the property of existence. By the same token,

$$\text{Winged horses do not exist} \tag{12.3}$$

would be treated as the negation of a subject/predicate sentence: (12.3) would be analyzed in such a way that the term “Winged horses” refers to some things and the sentence says about them that they lack the property of existence. So (12.1) would be expressed in a first-order language as

$$(\exists x)(x \text{ is a winged horse} \wedge x \text{ exists}) \tag{12.4}$$

and (12.3) would be expressed as

$$(\exists x)(x \text{ is a winged horse} \wedge \neg x \text{ exists}). \tag{12.5}$$

But, on Russell's view, this is wrong. Neither (12.1) nor (12.3) involves any reference to things *per se*. There are no winged horses, and so there are no such

objects to refer to. Instead, (12.1) says there are things of a certain sort and (12.3) denies that there are things of a certain sort. So, the proper first-order interpretations of (12.1) and (12.3) are, respectively,

$$(\exists x)(x \text{ is a winged horse}) \quad (12.6)$$

and

$$\neg(\exists x)(x \text{ is a winged horse}). \quad (12.7)$$

What about a sentence like

$$\text{Pegasus exists} \quad (12.8)$$

which contains a singular term that *prima facie* refers to an object? Russell's view is that the name in (12.8) is not really serving as a genuine referential term and so the sentence should not be assigned the logical form of an atomic sentence, $F(a)$; rather we are to replace the name by a description or, as Quine (1948) later suggested, simply create one,

$$(\exists x)(x \text{ pegasizes}), \quad (12.9)$$

where *pegasizes* or *is-identical-with-Pegasus* is an artificial predicate, true only of Pegasus. Predicates do not carry the same sort of existential commitment that referring expressions appear to, and so switching to a predicate construction, as Quine recommends, provides us with an escape from the Paradox.

The Deflationist *reductio* entitles us to conclude that either it cannot be the case that "Winged horses" is the subject of (12.1) or it cannot be the case that "exists" is the predicate. The Russellian Deflationist, however, appears to infer something stronger, namely, that "Winged horses" is not the subject and "exists" is not the predicate. This is unnecessarily strong. Let us look at the four possible analyses of (12.1)

1. "Winged horses" is subject; "exists" is predicate.
2. "Winged horses" is subject; "exists" is not predicate.
3. "Winged horses" is not subject; "exists" is predicate.
4. "Winged horses" is not subject; "exists" is not predicate.

The *reductio* only eliminates 1. Item 2 has no plausibility because one cannot have a subject without a predicate, so it too can be eliminated. This leaves 3 and 4. "Winged horses" cannot be the subject, whether it is true or false that "exists" is the predicate. So, there is little reason to be found in The Paradox of NonBeing for rejecting "exists" as a predicate.

Unfortunately, discussions of the Paradox of NonBeing have become intertwined with the notorious Ontological Argument for the existence of God. There is a widely entrenched belief that, to avoid the conclusion of the Ontological Argument, one

must deny that *existence* is a property of objects. The Paradox of NonBeing has been viewed as confirming the logical correctness of this view and taking it one step further. Indeed, it has become standard to argue that “Winged horses” cannot be the subject of (12.1) *because* “exists” is not the predicate. We quote from the very influential explanation and defense of Russell’s theory from Ryle (1932):

Since Kant, we have, most of us, paid lip service to the doctrine that ‘existence is not a quality’ and so we have rejected the pseudo-implication of the ontological argument: ‘God is perfect, being perfect entails being existent, God exists.’ For if existence is not a quality, it is not the sort of thing that can be entailed by a quality.

But until fairly recently it was not noticed that if in ‘God exists’ ‘exists’ is not a predicate (save in grammar), then in the same statement ‘God’ cannot be (save in grammar) the subject of predication. The realization of this came from examining negative existential propositions like ‘Satan does not exist’ or ‘Unicorns are nonexistent’. If there is no Satan, then the statement ‘Satan does not exist’ cannot be about Satan in the way in which ‘I am sleepy’ is about me. Despite appearances the word ‘Satan’ cannot be signifying a subject of attributes. (p. 42)

But Ryle is wrong. Kant (1781) denied that “exists” was a predicate that enlarged the concept; he never denied that it was a predicate.

The idea of translating (12.1) into first-order logic as (12.6) appears to have been proposed first in the modern literature by Frege (1884): in his vocabulary, (12.1) is a statement about a concept, for it says that something falls under the concept *winged horse*.¹ Frege’s analysis has been echoed by many others who followed him, and it has been supposed that the logical work of the grammatical predicate “exists” is exhausted by the existential quantifier, $\exists$. There are two ways of understanding the view just expressed. First, “exists” is not a predicate; an existential assertion is to be viewed as an existentially quantified sentence following the translation scheme just indicated (i.e., replace (12.1) by (12.6), replace (12.8) by (12.9)). Second, whether or not “exists” is a predicate, the translation indicated is appropriate and exhaustive of the content of an existential claim.

But neither of these views is correct. There is no reason why we cannot include a predicate “*x* exists” in a classical first-order language if we hold that it is a predicate true of everything. Denials of existence, of course, turn out to be false. But in a very important sense, they should: classically everything exists, and so it is false that there are things that don’t exist. Where the situation gets tricky, as in our use of names and descriptions in later chapters, we need a more complicated logical story about the role of these expressions. Russell’s, among others, will do. But, nothing in this more complicated story requires that we abandon the idea that “exists” is a predicate.

So, there is little reason on the Deflationist view to deny that “exists” is a predicate. Is this predicate, however, redundant in classical first-order logic? More

¹ Frege never provided an analysis for *singular* existentials like (12.8). Carnap (1947) introduced the notion of an *individual concept* to this end; Quine (1948), working within the alternate tradition of handling singular terms devised by Russell (1905), invented the artificially constructed predicate used in (12.9).

precisely, is the work done by “exists” exhausted by the existential quantifier? The answer is “No.” For there is no way of saying “Something exists” or “Everything exists” unless there is a predicate—either primitive or defined—available to do the work of “exists”: it will make no sense simply to use the quantifiers.

Our discussion of the Deflationist response to the Paradox of NonBeing has focused on the idea that in denying that something exists, we need refer to it. But we have still to look at the other assumption in the Deflationist position, viz., that we can only refer to things that exist.

12.5 Parmenides’ Principle

Premise (B), from Sect. 12.3, carries the burden in the argument. We call it *Parmenides’ Principle*, and we repeat it for convenience.

Parmenides’ Principle Things which do not exist cannot be referred to or mentioned; no statement can be about them.

Parmenides’ Principles is widely espoused. Plato, as we just saw, accepted it. So too did Frege (1884) and Russell (1905). More recently, Strawson (1950) says that one cannot refer to fictional or imaginary creatures. Searle (1968) takes “Whatever is referred to must exist” as a basic axiom of the theory of reference. And Quine (1948) holds that a singular term denotes if, and only if, the term can be replaced by a bound variable.

Still, Parmenides’ Principle is not obviously true. In fact, on our ordinary understanding of the key terms “exist” and “refer to,” it cannot plausibly be defended. For we do assert, for example, that Santa Claus lives at the North Pole, that Pegasus was captured by Bellerophon, that phlogiston does not exist. In each case, we talk about, respectively, Santa Claus, Pegasus, Bellerophon, and phlogiston—entities which no one today believes to exist, yet communication and intelligibility are not impaired. We can perfectly well identify what we are talking about. Surely, nobody confuses Santa Claus with either the archangel Gabriel or the novelist Marcel Proust, yet we all recognize that Santa Claus, also known as Saint Nick, is none other than Kris Kringle.

Ordinary language considerations aside, Parmenides’ Principle is objectionable at a more profound level. Wittgenstein (1922), in an oft-quoted remark, warned of attempting to circumscribe the conceivable:

... in order to be able to set a limit to thought, we should have to find both sides of the limit thinkable (i.e., we should have to be able to think what cannot be thought). (p. 3)

Parmenides’ Principle, which appears to speak about what cannot be spoken about, commits just such a blunder. This is precisely the problem the Eleatic Stranger puts his finger on in the passage from the *Sophist* quoted in Sect. 12.3.

Let’s go through the problem slowly. In the original form as Cartwright phrased it, the condition on reference was “Things which do not exist cannot be referred

to or mentioned; no statement can be about them.” Stated this way, **(B)** is self-defeating: the very making of the claim falsifies what is being claimed. Consider the second half of **(B)**, which says that no statement can be about things which do not exist. If no statement can be about things which do not exist, then, in particular, **(B)** cannot be about things which do not exist. But certainly, premise **(B)** tells us what no statement can be about: things which do not exist. The same self-defeating quality also infects the first part of **(B)**. **(B)** says that the expression “things which do not exist” cannot refer to things which do not exist. Yet **(B)** does succeed, via this expression, in indicating what it is that cannot be referred to. And so, as we argued before, premise **(B)** refers to things which do not exist.

There does, however, seem to be a simple way out of this bind. Admit that **(B)** is ill-constructed and give it up. Instead of specifying what cannot be referred to, rephrase the condition so as to specify what can be referred to. This can be done easily. Recall the conditional form of premise **(B)** which we used for testing the validity of the paradox:

(D) If John refers to or mentions something, Ks, then Ks exist.

(D) is not, in any straightforward sense, about things which do not exist, and there does not seem to be any reference to things which do not exist; So, **(D)** is not self-defeating like **(B)**. And, most important, **(D)** says just what was intended in **(B)**, namely, that existence is a necessary condition for referring. The dilemma we posed in the previous paragraph has, therefore, been sidestepped: we do not have to specify what cannot be referred to, and so we do not have to refer to things which do not exist. The Deflationist position, appears to be consistent.²

12.6 Inflationists

The author of *The Principles of Mathematics*, Bertrand Russell, famously urged: “[W]hat does not exist must be something; or it would be meaningless to deny its existence; and hence we need the concept of being as that which belongs to the nonexistent.” Russell explains:

Being is that which belongs to every conceivable term, to every possible object of thought—in short, to everything that can possibly occur in a proposition, true or false, and to all such propositions themselves. Being belongs to whatever can be counted. If A be any term that can be counted as one, it is plain that A is something, and therefore that A is. ‘A is not’ must always be either false or meaningless. For if A were nothing, it could not be said not to be; ‘A is not’ implies that there is a term whose being is denied, and hence that A is. Thus unless ‘A is not’ be an empty sound, it must be false—whatever it may be, it certainly is. Numbers, the Homeric gods, chimeras, and four-dimensional spaces all have being, for if these were not entities of a kind, we could make no propositions about them. Thus being

² Thanks to Hanson (2006), we have clarified our narrative here in the second edition.

is a general attribute of everything, and to mention anything is to show that it is. (Russell 1938, p. 449)

Russell allows us to deny the existence of a thing; it is *being* we cannot deny. *Being* is the more general attribute that belongs to everything; *existence* holds of only some beings.

Russell, therefore accepts premise (A) of the Paradox (Sect. 12.3): he agrees that in order to deny the existence of an object, we must refer to it. But he rejects premise (B), supplanting it instead with

(E) Things which do not have being cannot be referred to or mentioned; no statement can be about them.

Existence is not a necessary condition for referring, but *Being* (something) is. There is an analogue of the original paradox for the Inflationist, the upshot of which is that there can be no true, meaningful denials of *Being*.

Russell's inflationism appears crass and dogmatic by comparison with the more subtle views of Alexius Meinong, the Austrian philosopher from whom he took his cue. To be sure, Meinong (1904) expressed his contempt for this prejudice for the actual:

[T]he totality of what exists, including what has existed and will exist, is infinitely small in comparison with the totality of the objects of knowledge. This fact easily goes unnoticed, probably because the lively interest in reality which is part of our nature tends to favor that exaggeration which finds the non-real a mere nothing . . . or, more precisely, which finds the non-real to be something for which science has no application or at least no application of any worth. (Chisholm 1960, 82)

And his own position is specifically provocative:

Those who like paradoxical modes of expression could very well say: 'There are objects of which it is true to say that there are no such objects'. (Chisholm 1960, 83)

But his suggestion is no mere terminological trick to inflate the universe. Meinong spoke of objects as beyond *Being* and *NotBeing*, a point embodied in his Principle of the Independence of *Being-so* [Sosein] from *Being* [Sein]:

[T]he *Sosein* of an Object is not affected by its *Nichtsein*. The fact is sufficiently important to be explicitly formulated as the principle of the independence of *Sosein* from *Sein*. The area of applicability of this principle is best illustrated by consideration of the following circumstance: the principle applies, not only to Objects which do not exist in fact, but also to Objects which could not exist because they are impossible. Not only is the much heralded gold mountain made of gold, but the round square is as surely round as it is square. (Chisholm 1960, 82)

His goal, then, is not the overpopulated unruly slum of a universe Quine (1948) ridiculed. Meinong's point is that an object can have properties even if it does not exist—indeed, even if it could not exist. Whether or not an object has such-and-such properties is *independent* of whether it exists. And this is just the semantic principle we chose in Sect. 8.7 when dealing with varying domain semantics: $F(x)$ is assigned a truth value at a world in which the value of x does not exist. We do appear to be able to identify and re-identify things that do not exist—Pegasus, Sherlock Holmes,

Santa Claus. These do not appear to be utterly unreal. It is not as if one is faced with emptiness, a void: there is something on which to focus our attention and about which we can converse intelligently.

Meinong was a student of Brentano. He was influenced by Brentano's view that all mental attitudes are directed to objects—in some cases, to objects that do not exist. Meinong's task was to characterize the "intensional objects" Brentano had posited as that to which the mind is directed. An individual who is thinking of Pegasus is thinking of something. To be sure, Pegasus does not exist; but the thought is directed toward him, and so he must be something. I can entertain the thought (as all, by and large, admit) that the great racehorse Man-O-War had wings: here, my thought is directed at a nonexistent state of affairs, namely, Man-O-War's having wings. Why can I not similarly entertain the thought that Pegasus ran in the Kentucky Derby? Here my thought is also directed at a nonexistent state of affairs, namely, Pegasus's having run in the Kentucky Derby. Meinong's proposal is that referring is intensional. Unlike the verb "hit," where if it is true that John hit Jim, Jim must exist, John can refer to Jim even though Jim does not exist. "Refer," then, is like "worship," "assume," "postulate": the object of the verb need not exist.

Russell appears to be invoking the following principle.

Russell's Schema: If 'A' is a meaningful singular term, A must be.

Essence does not entail *Existence*, as one tradition would have it, but it does entail *Being* on this view. But Meinong invokes something a bit stronger, an *Unrestricted Comprehension Schema for Singular Terms*, which we might put as follows.

Meinong's Schema: If 'the A' is a meaningful singular term, the A must be A.

Routley (1980) calls this "the characterization postulate." So stated, the comprehension schema is highly dubious.

Meinong's position has come down to us by and large as a prime example of philosophical foolishness, a target of derision ever since Russell regained his "robust sense of reality" and turned on it. It has found very few adherents. (Although Parsons (1985) and Routley (1980) have attempted to resuscitate the view.) What we have tried to do in these few paragraphs is make the motivation clearer. The problem Meinong addressed is real, and requires real solutions. It cannot be swept under the carpet. But the particular solution he offered has insuperable difficulties, which Russell himself pointed out.

Russell (1905) claimed that it violated the Law of Noncontradiction. According to Meinong, any significant denial of existence requires that that which is denied existence *be* something. Consider the claim that there is no least prime number in the open interval (19, 23). This assertion is meaningful, even true; so there must be a least prime number in the interval (19, 23), else it would make no sense to deny its existence. But which number could this be? The only integers in the interval are 20, 21 and 22, and none of these is prime. We appear to be committed to saying that there is a prime number in the interval (19, 23) ...and also that there is no prime number in the interval (19, 23). To be sure, Meinong is aware of this result: he himself notes that the round square is as much round as it is square, which is

to say that it is as much round as it is not round. How are we to make sense of this? Impossible objects are objects that cannot possibly be—and therefore cannot possibly be so or not be so.

Meinong, when faced with this difficulty, remained steadfast in his position and believed he could defend it. Russell, who pointed out the difficulty to him, was convinced this philosophical battle was over. He went over the hill and joined the Deflationists.

We adopt a position less radical than either of these two men, while retaining the flavor of the Meinongian approach. We remain agnostic about the two philosophical positions, but we provide a formal language and a formal semantics that enables one to designate nonexistent, but possible objects. We distinguish terms that purport to designate nonexistent objects, like “the golden mountain,” from terms that fail to designate at all, like “the round square.” But the discussion of this distinction will have to await the introduction of names and descriptions into our logical vocabulary, and the machinery of predicate abstraction.

Exercises

Exercise 12.6.1 “There is nothing inflationary about inflationism, since inflationism postulates no more existents than does deflationism.” Discuss.

12.7 Unactualized Possibles

Quine (1948) presents the following famous criticism of the fictitious philosopher Wyman (a stand-in for Meinong) who believes in *unactualized possibles*:

Wyman’s slum of possibles is a breeding ground for disorderly elements. Take, for instance, the possible fat man in that doorway; and, again, the possible bald man in that doorway. Are they the same possible man, or two possible men? How do we decide? How many possible men are there in that doorway? Are there more possible thin ones than fat ones? How many of them are alike? Or would their being alike make them one? Are no *two* possible things alike? Is this the same as saying that it is impossible for two things to be alike? Or, finally, is the concept of identity simply inapplicable to unactualized possibles? But what sense can be found in talking of entities which cannot meaningfully be said to be identical with themselves and distinct from one another? These elements are well-nigh incorrigible. By a Fregean therapy of individual concepts, some effort might be made at rehabilitation; but I feel we’d do better simply to clear Wyman’s slum and be done with it. (p. 4)

We do not wish to defend the Meinongian view that the very meaningfulness of a singular term requires that there *be* an object the singular term stands for. We do not wish to endorse this reason for introducing unactualized possibles. But, we do use the notion of unactualized possibles in our construction of varying domain semantics, and we need to say something to defend the coherence of the idea.

The proper response to Quine's story is that the doorway is empty. There is nothing in the doorway. There is no fat man in the doorway. There is no thin man in the doorway. There is no possible fat man in the doorway. There is no possible thin man in the doorway. There is just nothing in the doorway.

It makes no sense to place unactualized possibles inside the actual world. That is the point of saying that they are unactualized.

Wyman thinks that there is no *actual* man in the doorway but that there is a *possible* one in the doorway. What does this mean? It could mean two distinct things. On the one hand, it could mean there is a man who is actual and who is not in the doorway, but who, in another possible world, is in the doorway. Julius Caesar is not in the doorway; but in another possible world, he is. On the other hand, it could mean that there is a possible man, e.g., Sherlock Holmes—an element in the domain of another possible world but not the actual one—who, in that world, is in the doorway. In neither case, however, do we have *in this world* anyone in the doorway.

Suppose we consider "the future man in the doorway." (Of course, it is always open to someone to say that this is an improper description insofar as it most likely fails to pick out a unique individual. Ignore this for the following discussion.) We might mean the actual man who in the future is in the doorway; we might mean the man who is not actual now but will be in the future who is in the doorway. In each case, there is no man in the doorway but there will be; in the first case, it is a man who exists now, but in the second case, it is a man who will exist in the future.

The temporal analogy gives a better handle on the question "How many?" In the first case, that depends upon which men will in the future appear in the doorway. It is doubtful whether we can make that determination now, but in the future, long after everyone who is now alive is dead, such a determination can be made. We go back to the year 2022 (when this second edition was written) and see which men alive in 2022 subsequently appeared in the doorway. The question is understandable and can in principle be answered. How about with the second case? Here we have no clear answer. The problem is that we never reach a point at which we look back and count. There is always another moment in time. It could be, the correct answer is that there are an unlimited number of such men.

An unactualized possible does not exist in this world. But this does not mean that objects cannot exist in more than one possible world. We can speak of what *this* individual is like in another world. Similarly, we can speak of what individual in another world is in *this* doorway. It is not as if we have a shadowy individual in *this* doorway (Quine's "possible fat man in the doorway"). Once again, in the actual world, there is nothing in the doorway. In another world, there is.

Nonetheless, there is considerable antipathy toward the idea that there are things that exist in other possible worlds but not the actual world. As such the disagreement between Inflationists and Deflationists has morphed into a disagreement between Actualists and Possibilists. The issue that separates them is no longer primarily about what one can refer to; rather it is about what there is. Possibilists hold that there are objects that could exist but do not. Actualists, on the other hand, deny that there are things that could exist but don't actually exist. One major point of contention between Actualists and Possibilists is the Barcan Formula—which, as

we explained earlier in Definition 8.10.1, is actually a schema:

$$\Diamond \exists x \Phi \supset \exists x \Diamond \Phi$$

or equivalently,

$$\forall x \Box \Phi \supset \Box \forall x \Phi.$$

The Barcan Formula says, in effect, that if Δ is accessible from Γ , then the domain of Δ is a subset of the domain of Γ . There is nothing that exists in a possible world that does not already exist in the actual one. So if, for example, in another possible world there exists a unicorn, then in the actual world, there must exist something that could be a unicorn. Actualists by and large accept the Barcan Formula; Possibilists reject it. The Barcan Formula is not valid on Varying Domain semantics; it is on Constant Domain semantics. For this reason, the former has been termed Actualist Quantification and the latter Possibilist Quantification. But because, as we saw in Sect. 8.9, there is no logical significance in whether we choose Actualist or Possibilist Quantification, one should be careful about this connection to the Actualist/Possibilist controversy. (For a discussion of the issues, see Menzel (2022), Bennett (2005, 2006), Linsky and Zalta (1996), Nelson and Zalta (2009). Williamson (2013) has claimed that the terms of the dispute are confused, and he has suggested the dispute should rather be framed as between Necessitists and Contingentists.)

Exercises

Exercise 12.7.1 Construct a varying domain **K** model with distinct objects a and b so that, at a particular world Γ , a and b satisfy exactly the same formulas, provided those formulas do not contain $=$, and such that at Γ , $E(x)$ is true if the value of x is a , and false if the value of x is b .

Exercise 12.7.2 The solution to Quine's puzzle works best for varying domain semantics. Is there a comparable story to be told for constant domain semantics?

12.8 Barcan and Converse Barcan, Again

We have concluded the discussion of philosophical issues that began with Sect. 12.3. Now we turn to more formal matters, concerning the behavior of models and tableaux.

Monotonicity (objects that exist continue to do so in alternate worlds) and anti-monotonicity (no additional objects exist in alternate worlds) are important

existence assumptions. These were investigated in Sect. 8.10. The two together give us locally constant domain models (Definition 8.10.9), and these have the same set of validities as constant domain models (Proposition 8.10.10). That is, we can think of varying domain semantics, restricted to monotonic, anti-monotonic models, essentially as constant domain semantics.

Monotonicity and anti-monotonicity are semantic conditions, but we saw in Sect. 8.10 that they correspond to something syntactic as well. Monotonicity of a frame is equivalent to the validity of all instances of the Converse Barcan formula, Definition 8.10.2, while anti-monotonicity is equivalent to the validity of all instances of the Barcan formula. But these syntactic versions have their drawbacks. Both Barcan and the Converse Barcan are schemas, with infinitely many formulas as instances. But now we have a richer language than we had in Chap. 8—equality has been added—and it turns out that these infinitely many instances can be replaced by single formulas. Although these formulas may not be sentences—they may involve free variables—they correspond to intuition quite closely.

We begin with the Converse Barcan formula, for which there are actually two equivalents, only one of which involves a free variable. Recall that the Converse Barcan formula corresponds to monotonicity, and this says that whatever exists continues to do so in all alternative worlds. At each world the actualist quantifier ranges over what exists at that world. So using our existence formula (which is defined in terms of equality) we can express monotonicity very simply with the single sentence $(\forall x)\Box E(x)$, Definition 12.1.2.

There is an alternative way of capturing monotonicity syntactically. Since it involves free variables, it is somewhat less handy than the sentence we just gave, but it provides some additional insights. Here is our second way of capturing monotonicity: $E(x) \supset \Box E(x)$.

The following theorem says we have succeeded in what we set out to do: express monotonicity syntactically.

Theorem 12.8.1 *Let $\mathcal{F} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ be a varying domain skeleton. The following are equivalent:*

1. $\mathcal{F}$ is monotonic.
2. The Converse Barcan formula is valid in every model based on $\mathcal{F}$.
3. $E(x) \supset \Box E(x)$ is valid in every normal model based on $\mathcal{F}$.
4. $(\forall x)\Box E(x)$ is valid in every normal model based on $\mathcal{F}$.

Proof The equivalence of items 1 and 2 was shown earlier in Chap. 8, as Proposition 8.10.6. The equivalence of item 1 and item 3 is rather straightforward, and we leave it as an exercise. That item 2 implies item 4 is easy: one instance of the Converse Barcan formula is $\Box(\forall x)E(x) \supset (\forall x)\Box E(x)$, but we also have the validity of $\Box(\forall x)E(x)$ (in Example 12.2.1 we gave a tableau proof of $(\forall z)E(z)$, and the validity of its necessitation follows). Showing that item 4 implies item 2 is most easily done using tableaux, and we leave it as an exercise as well.

There is an important observation to be made at this point. Given a frame, the interpretation of the equality symbol in all normal models based on this frame will

be the same since the equality symbol is always interpreted by the equality relation on the frame. It follows that a formula involving only the equality relation is valid in *some* model based on a frame if and only if it is valid in *every* model based on that frame. This is the case for $E(x) \supset \Box E(x)$ and $(\forall x)\Box E(x)$, and is a most important feature.

Next we turn to the Barcan formula. This time we have only a single equivalent. The Barcan formula corresponds to anti-monotonicity: anything that exists at a world alternative to this one also exists here. The open formula $\Diamond E(x) \supset E(x)$ expresses this very nicely. And the following says formally that it does so. Part of it was proved in Chap. 8 as Proposition 8.10.8. The rest is left as an exercise.

Theorem 12.8.2 *Let $\mathcal{F} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ be a varying domain skeleton. The following are equivalent:*

1. $\mathcal{F}$ is anti-monotonic.
2. The Barcan formula is valid in every model based on $\mathcal{F}$.
3. $\Diamond E(x) \supset E(x)$ is valid in every normal model based on $\mathcal{F}$.

Exercises

Exercise 12.8.1 Show the equivalence of items 1 and 3 in Theorem 12.8.1.

Exercise 12.8.2 Give a varying domain tableau proof of the following.

$$(\forall x)\Box E(x) \supset [\Box(\forall x)\Phi(x) \supset (\forall x)\Box\Phi(x)]$$

Exercise 12.8.3 Show the equivalence of items 1 and 3 in Theorem 12.8.2.

12.9 Using Validities in Tableaus

Almost all tableau proofs we have given have been direct—they did not make use of assumptions. In Sect. 7.3 we did discuss how to use local and global assumptions in propositional tableau proofs, but for the most part this has played little role so far. Now such ideas become particularly useful because, as we have seen in the previous section, there are formulas that capture monotonicity and anti-monotonicity, and these are important semantic notions. We only need *global* assumptions in tableaus now, but we will need to make use of open formulas, which hitherto have played no role in tableaus. For convenience, we start from the beginning here—review of Sect. 7.3 would be nice, but is not necessary.

Definition 12.9.1 (Using a Closed Global Assumption) To use a *closed* formula Φ as a global assumption in a tableau proof the rule is: $\sigma \Phi$ can be added to any branch, for any prefix σ that occurs on the branch.

This means we can have a version of tableaux incorporating *monotonicity* by using the varying domain rules and taking $(\forall x)\Box E(x)$ as a global assumption.

Example 12.9.2 Here is a tableau proof, using the varying domain **K** rules, of $\Box(\exists x)\Diamond A(x) \supset \Box\Diamond(\exists x)A(x)$, taking $(\forall x)\Box E(x)$ as a global assumption. Thus in effect it is a monotonic proof of the formula.

- 1 $\neg[\Box(\exists x)\Diamond A(x) \supset \Box\Diamond(\exists x)A(x)]$ 1.
- 1 $\Box(\exists x)\Diamond A(x)$ 2.
- 1 $\neg\Box\Diamond(\exists x)A(x)$ 3.
- 1.1 $\neg\Diamond(\exists x)A(x)$ 4.
- 1.1 $(\exists x)\Diamond A(x)$ 5.
- 1.1 $\Diamond A(p_{1.1})$ 6.
- 1.1.1 $A(p_{1.1})$ 7.
- 1.1.1 $\neg(\exists x)A(x)$ 8.
- 1.1 $(\forall x)\Box E(x)$ 9.
- 1.1 $\Box E(p_{1.1})$ 10.
- 1.1.1 $E(p_{1.1})$ 11.
- 1.1.1 $\neg A(p_{1.1})$ 12.

Items 2 and 3 are from 1 by a Conjunctive Rule; 4 is from 3 by a Possibility Rule; 5 is from 2 by a Necessity Rule; 6 is from 5 by an Existential Rule; 7 is from 6 by a Possibility Rule; 8 is from 4 by a Necessity Rule; 9 is our global assumption; 10 is from 9 by a Universal Rule; 11 is from 10 by a Necessity Rule; and 12 is from 8 and 11 by the derived Parameter Existence Rule, Definition 12.2.2.

Using an *open* formula as a global assumption is almost as easy. Remember, a free variable can represent anything in the domain of the model.

Definition 12.9.3 (Using an Open Global Assumption) To use an open formula $\Phi(x)$ as a global assumption in a tableau proof the rule is: $\sigma \Phi(p_\tau)$ can be added to any branch, for any prefix σ and any parameter p_τ that occur on the branch.

Now we can get the effect of anti-monotonicity by using the varying domain rules and taking $\Diamond E(x) \supset E(x)$ as an assumption.

Example 12.9.4 Here is a tableau proof, using the varying domain **K** rules, of an instance of the Barcan formula, $(\forall x)\Box A(x) \supset \Box(\forall x)A(x)$, taking $\Diamond E(x) \supset E(x)$ as an open global assumption. In effect, it is a tableau verification of part of Theorem 12.8.2.

- 1 $\neg[(\forall x)\Box A(x) \supset \Box(\forall x)A(x)]$ 1.
- 1 $(\forall x)\Box A(x)$ 2.
- 1 $\neg\Box(\forall x)A(x)$ 3.
- 1.1 $\neg(\forall x)A(x)$ 4.
- 1.1 $\neg A(p_{1.1})$ 5.
- 1 $\Diamond E(p_{1.1}) \supset E(p_{1.1})$ 6.

Items 2 and 3 are from 1 by a Conjunctive Rule; 4 is from 3 by a Possibility Rule; 5 is from 4 by an Existential Rule; 6 is by our global assumption;

At this point the tableau branches, using item 6. The left branch contains $1 \neg\Diamond E(p_{1.1})$, from which we get $1.1 \neg E(p_{1.1})$, and this branch closes immediately using the Parameter NonExistence derived rule, Definition 12.2.3. The right branch continues as follows.

- 1 $E(p_{1.1})$ 7.
- 1 $\Box A(p_{1.1})$ 8.
- 1.1 $A(p_{1.1})$ 9.

Item 8 follows from items 2 and 7 using the Parameter Existence derived rule, Definition 12.2.2; 9 follows from 8 by a Necessity Rule.

Finally, by using both $(\forall x)\Box E(x)$ (or $E(x) \supset \Box E(x)$) and $\Diamond E(x) \supset E(x)$ as global assumptions, varying domain tableau machinery allows us to construct what are, in effect, constant domain proofs. We do not recommend this—the constant domain tableau rules we gave earlier are simpler to use. The point is simply that varying domain rules can be made to do constant domain work, just as constant domain rules can simulate varying domain arguments. On purely formal grounds, neither version has primacy.

Exercises

Exercise 12.9.1 Give a varying domain **K** proof of

$$\Box(\forall x)(\exists y)\Diamond R(x, y) \supset (\forall x)\Box\Diamond(\exists y)R(x, y)$$

using $(\forall x)\Box E(x)$ as a global assumption.

12.10 Tableaus Imitate Tableaus

Using the machinery given in the previous section, 12.9, varying domain tableaus can be used to establish constant domain validities. We simply need to take

the closed formula $(\forall x)\Box E(x)$ and the open formula $\Diamond E(x) \supset E(x)$ as global assumptions in our varying domain tableaux; Definitions 12.9.1 and 12.9.3. As it happens, the other way around is also possible; that is, one can use constant domain tableaux to establish varying domain validities. Here's how.

In Sect. 8.9 the relativization of a formula to an existence predicate was introduced, Definition 8.9.1, and it was established in Proposition 8.9.2 that this turned a varying domain validity problem into a constant domain one. Based on that Proposition, we can use constant domain tableaux to establish varying domain validities. First, we must work with formulas relativized to a primitive, not defined, existence predicate, $\mathcal{E}$, and second, we must somehow guarantee that this existence predicate 'thinks' each possible world is non-empty. We can do this easily by adopting the following rule.

Definition 12.10.1 (Non-empty Domains Rule) $\sigma (\exists x)\mathcal{E}(x)$ can be added to the end of any tableau branch on which the prefix σ occurs.

We present some examples showing how this works.

Example 12.10.2 The formula

$$[(\forall x)\Diamond P(x) \wedge (\exists x)\Box Q(x)] \supset (\exists x)\Diamond[P(x) \wedge Q(x)]$$

is valid in varying domain **K**. Of course it trivially follows that it is valid in constant domain **K**, and hence provable using the constant domain rules. This is not very interesting. We leave it to you to establish the *varying* domain validity by constructing the relativization of this formula with $\mathcal{E}$, and then proving the result using constant domains tableaux. As it happens, the Non-Empty Domains Rule isn't needed.

What is more interesting than constructing proofs is extracting counter-examples from failed proof attempts. It is possible to extract *varying* domain counter-examples from failed constant domain tableau constructions for *relativized* formulas. We give an example, a version of the Barcan formula.

Example 12.10.3 $\Diamond(\exists x)P(x) \supset (\exists x)\Diamond P(x)$. The relativization of this is

$$\Diamond(\exists x)[\mathcal{E}(x) \wedge P(x)] \supset (\exists x)[\mathcal{E}(x) \wedge \Diamond P(x)].$$

A failed proof attempt using constant domain **K** rules, and the Non-Empty Domain Rule, is displayed in Fig. 12.1. Only item 12 really needs specific comment, but here are all the steps, for the record. 2 and 3 are from 1 by an Implication Rule, 4 is from 2 by a Possibility Rule, 5 is from 5 by an Existential Rule, 6 and 7 are from 5 by a Conjunctive Rule, and 8 is from 3 by a Universal Rule. Next, 9 and 10 are from 8 by a Disjunctive Rule, 11 is from 10 by a Necessity Rule, and the right branch is closed by 7 and 11. On the left branch, we have that something exists with prefix 1.1 by 6 (and so in the model, the corresponding possible world domain will be non-empty). There is nothing similar for prefix 1, so 12 is added using the Non-Empty Domain

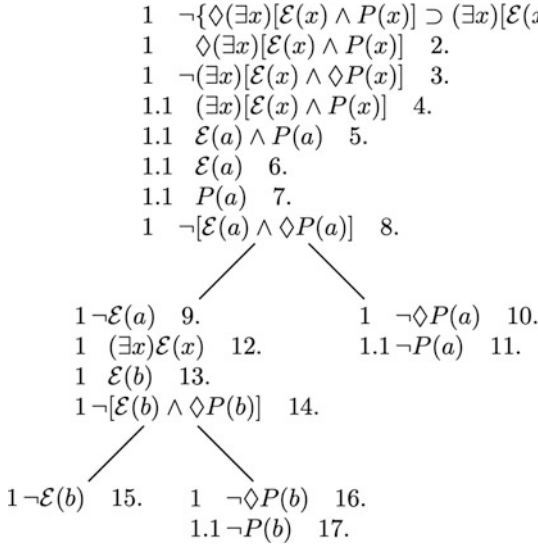


Fig. 12.1 Failed constant domain tableau proof

1. $\Diamond E(x) \supset E(x)$ anti-monotonicity
2. $\Box\Diamond E(x) \supset \Box E(x)$ necessitation
3. $E(x) \supset \Box E(x)$ using $\Phi \supset \Box\Diamond\Phi$

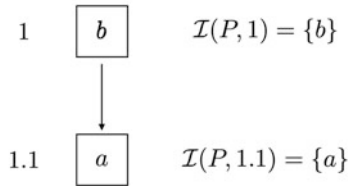


Fig. 12.2 A varying domain counter model

Rule, then 13 follows by an Existential Rule, and 14 follows from 3 by a Universal Rule. Finally, 15 and 16 are from 14 by a Disjunctive Rule, and 17 is from 16 by a Necessity Rule.

The leftmost branch is closed by 13 and 15, but the center branch is not closed. We use it to construct a varying domain counter-model, which is shown in Fig. 12.2. First, the possible worlds are the prefixes appearing on the branch, 1 and 1.1, with 1.1 accessible from 1. The domain of world 1 is $\{b\}$ because of 13, and of world 1.1 is $\{a\}$ by 6. Finally, the interpretation of P at world 1 is empty, and at world 1.1 is $\{a\}$.

12.11 On Symmetry

This section consists of a small remark, but it is of some technical interest. If the accessibility relation of a frame is symmetric, monotonicity and anti-monotonicity are equivalent—either implies the other. To say monotonicity and anti-monotonicity are equivalent is to say each instance of the Barcan formula is derivable from some instances of the Converse Barcan formula, and each instance of the Converse Barcan formula is derivable from some instances of the Barcan formula.

If we use the Barcan/Converse Barcan equivalents involving $E(x)$, the equivalence is straightforward to verify. Recall that to say a frame has a symmetric accessibility relation is equivalent to saying that all instances of the schema $\Phi \supset \Box\Diamond\Phi$ are valid in it, or equivalently, all instances of $\Diamond\Box\Phi \supset \Phi$ are valid. Now we have the following informal argument (formalizable axiomatically, however).

1. $\Diamond E(x) \supset E(x)$ anti-monotonicity
2. $\Box\Diamond E(x) \supset \Box E(x)$ necessitation
3. $E(x) \supset \Box E(x)$ using $\Phi \supset \Box\Diamond\Phi$

Thus assuming a syntactic equivalent to anti-monotonicity, plus a syntactic equivalent to symmetry, we can derive a syntactic equivalent to monotonicity. The other direction is equally easy.

Using the Barcan and Converse Barcan formulas directly is more work. Each instance of the Barcan formula is implied by a corresponding instance of the Converse Barcan formula, assuming symmetry. We give this as a tableau exercise below, Exercise 12.11.1. In the other direction, each instance of the Converse Barcan formula is implied by an instance of the Barcan formula as a global assumption, assuming symmetry. This is Exercise 12.11.2 below. In the first edition of this book we left it as an open problem how this could be established if we do not make use of equality. In fact, it can't. This was shown, in different ways, in Corsi (2002) and in Garson (2005).

Exercises

Exercise 12.11.1 Let $\widehat{B}$ be the formula $(\forall x)\Box A(x) \supset \Box(\forall x)A(x)$ (an arbitrary instance of the Barcan formula). Let $\widehat{CB}$ be the formula

$$\Box(\forall y)[(\forall x)\Box A(x) \supset \Box A(y)] \supset (\forall y)\Box[(\forall x)\Box A(x) \supset \Box A(y)]$$

(which is actually an instance of the Converse Barcan formula). Give a varying domain **B** proof of $\Box\widehat{CB} \supset \widehat{B}$. This shows that in **B**, every Barcan formula is implied by some Converse Barcan formula.

Exercise 12.11.2 Give a varying domain **B** proof of the Converse Barcan formula

$$(\exists x)\Diamond P(x) \supset \Diamond(\exists x)P(x)$$

using the (open) Barcan formula

$$\Diamond(\exists x)(x = y) \supset (\exists x)\Diamond(x = y)$$

as a global assumption.

Exercise 12.11.3 Beginning with the formula from Example 12.10.2, first relativize it using $\mathcal{E}$, then give a constant domain **K** tableau proof of the result.

Exercise 12.11.4 First, relativize the formula $(\forall x)\Box P(x) \supset \Box(\forall x)P(x)$ using $\mathcal{E}$. Then construct a failed constant domain tableau proof for the result, making use of the Non-Empty Domains Rule, and produce a varying domain counter-model from it.

References

- Bennett, K. (2005). Two axes of actualism. *The Philosophical Review*, 114(3), 297–326.
- Bennett, K. (2006). Proxy ‘actualism’. *Philosophical Studies*, 129, 263–294.
- Berlin, I. (1949–1950). Logical translation. *Proceedings of the Aristotelian Society*, NS, 50, 157–188.
- Carnap, R. (1947). *Meaning and necessity: A study in semantics and modal logic*. Chicago: University of Chicago Press.
- Cartwright, R. (1960). Negative existentials. *The Journal of Philosophy*, 57, 629–639. Reprinted in R. L. Cartwright (1987). *Philosophical essays*. Cambridge: MIT Press.
- Chisholm, R. M. (Ed.). (1960). *Realism and the background of phenomenology*. Glencoe: Free Press.
- Cornford, F. (1957). *Plato and Parmenides*. Indianapolis: Bobbs-Merrill.
- Corsi, G. (2002). A unified completeness theorem for quantified modal logics. *Journal of Symbolic Logic*, 67(4), 1483–1510.
- Frege, G. (1884). Dialogue with Pünjer on existence. In F. K. Hans Hermes & F. Kaulbach (Eds.), *Gottlob Frege, posthumous writings* (pp. 53–67). tr. Peter Long and Roger White (1979). Oxford: Basil Blackwell.
- Garson, J. W. (2005). Unifying quantified modal logic. *Journal of Philosophical Logic*, 34, 621–649.
- Hanson, W. (2006). The paradox of nonbeing. *Grazer Philosophische Studien*, 73, 205–219.
- Kant, I. (1781). *Critique of pure reason*. trans. Norman Kemp Smith (1964). London: Macmillan & Co.
- Linsky, B., & Zalta, E. N. (1996). In defense of the contingently nonconcrete. *Philosophical Studies*, 84(2/3), 283–294.
- Meinong, A. (1904). Über Gegenstandstheorie (The Theory of Objects). In J. A. Barth (Ed.), *Untersuchungen zur Gegenstandstheorie und Psychologie (Investigations in theory of objects and psychology)* (vol. 1, 6, pp. 1–51). [Translated as “The Theory of Objects” in Chisholm (1960, pp. 76–117).] Leipzig: Grazer Schule der Gegenstandstheorie und Experimentalpsychologie (Graz School of object theory and experimental psychology).

- Menzel, C. (2022). Actualism. In E. N. Zalta (Ed.), *The Stanford encyclopedia of philosophy*. <https://plato.stanford.edu/archives/sum-2022/entries/actualism>
- Nelson, M., & Zalta, E. N. (2009). Bennett and “proxy actualism”. *Philosophical Studies*, 142(2), 277–292.
- Parsons, T. (1985). *Nonexistent objects*. New Haven: Yale University Press.
- Quine, W. V. O. (1948). On what there is. *Review of Metaphysics*, 2. Reprinted in Quine, 1961.
- Quine, W. V. O. (1961). *From a logical point of view* (2nd. rev.). New York: Harper & Row.
- Routley, R. (1980). *Exploring Meinong’s Jungle and Beyond: An investigation of noneism and the theory of items*. Department of Philosophy, Monograph No. 3. Canberra: Australian National University.
- Russell, B. (1905). On denoting. *Mind*, 14, 479–493. Reprinted in Robert C. Marsh (Ed.), *Logic and knowledge: Essays 1901–1950, by Bertrand Russell*. London: Allen & Unwin (1956).
- Russell, B. (1938). *The principles of mathematics*. New York: Norton.
- Ryle, G. (1932). Systematically misleading expressions. *Proceedings of the Aristotelian Society*, 32. Reprinted in G. Ryle, *Collected Papers, Vol. II*, pp. 39–62. New York: Barnes & Noble (1971).
- Searle, J. (1968). *Speech acts*. Cambridge: Cambridge University Press.
- Strawson, P. F. (1950). On referring. *Mind*, 59, 320–344.
- Williamson, T. (2013). *Modal logic as metaphysics*. Oxford: Oxford University Press.
- Wittgenstein, L. (1922). *Tractatus Logico-Philosophicus*. tr. by D. F. Pears & B. F. McGuinness (1961). London: Routledge & Kegan Paul. *Annalen der Naturphilosophie*.

Part V

Predicate Abstraction and Scope

Presentations of first-order classical logic often allow constant and function symbols to appear in formulas. We are at an appropriate point for us to introduce them into our treatment of first-order modal logic as well. But doing so brings some complications with it. Since we have multiple possible worlds in our semantics, an issue arises that has no classical counterpart. Should a constant symbol act like a constant locally? That is, should it behave like a classical constant symbol in each possible world, though the value of the constant it represents might not be the same from world to world? Or should a constant symbol behave globally, representing the same constant value across the entire collection of possible worlds? Using standard terminology, should constant symbols be *non-rigid* or *rigid*? Both have their important uses. Rather than have two formal versions, we will introduce only non-rigid machinery. We can, if the need arises, think of rigid constant symbols as being non-rigid but just not happening to vary from world to world. The same issues apply to function symbols, but our initial discussions will be primarily about constant symbols since the basic modal aspects can be discussed most clearly in this simpler context. Nevertheless, function symbols will be introduced formally, and there will be examples that illustrate their uses.

It turns out that once we have non-rigid constant (and function) symbols, the world at which we evaluate them, where we ‘pin them down,’ becomes an important issue. Handling this requires that we pay careful attention to scope distinctions. We will see that, just as quantifiers have scopes in a formula, so do non-rigid constant and function symbols. Machinery for representing scopes will be introduced in this part. These issues of scope are among the most important things that we treat in the present work.

Chapter 13

Predicate Abstraction, Informally



Over the course of history, modal logic has earned a reputation for being difficult and confusing. To be sure, modal logic is more complex than classical logic, and to ease the way into the subject we have emphasized thinking in terms of possible worlds throughout this book. In this chapter, however, we will address some of the sources of difficulty. One of the main reasons for confusion is that the way in which we ordinarily express modal claims can frequently be interpreted—very naturally—in more than one way. This ambiguity is often hidden from view, and that is how even the most gifted of logicians can get tripped up. Accordingly we are going to spend a considerable amount of time in the current chapter identifying the most famous of these ambiguities, and explaining the notation that we have chosen to ensure that it is free from these problems.

The distinction we discuss here is that between *De Re* and *De Dicto* necessity. As the Latin indicates, it is a distinction that was identified in ancient times. We will go over many different examples so that you are thoroughly familiar with it and its problems. We will say something about the history of the distinction. We will identify it as an issue of scope, a familiar enough consideration in mathematics. And we will introduce a device into our modal notation, predicate abstraction, which is designed to assure an unambiguous notation, free from scope ambiguities.

13.1 Why Constants Should Not Be Constant

Classically constant symbols can represent mathematical objects, like the numbers 3 or π , but they can also represent things like Ghengis Khan or the concept of goodness—in fact, any object that might reside in the domain of a classical model. Similarly, function symbols represent mathematical functions like *sin* or *cos*, and also functions like *the-mother-of*. In a classical model whose domain is a set of people (including each person's mother), a particular function symbol can represent

the function that assigns, to each person in the domain, that person's mother. See almost any treatment of classical logic for the formal details—Fitting (1996) is a personal favorite.

Suppose we pick a constant symbol, c , with the informal intention that it mean *the tallest person in the world*. Of course c will designate different people at different times. Classically we regard the expression as incomplete, and we eliminate any ambiguity by completing it to something like *the tallest person in the world on January 1, 1900*. Let us use the constant symbol c' for the completed meaning. The designation of the constant c' is independent of the here and now. Let's see how this works in a modal setting, reading $\Box$ temporally, to mean *at all future times*, and thinking of possible worlds as instants of time. The tallest person at one instant of time might not be the same person as the tallest person at another instant of time, and so the incomplete c might very well designate different people at different instants of time. But the completed constant c' designates the same person at all times—whoever was tallest on January 1, 1900. Constant symbols that designate the same object in all possible worlds are called *rigid designators*. In our example, c' is a rigid designator but c is non-rigid.

Classically, such ambiguity is frowned upon and ambiguities are eliminated from logic. In everyday modal talk, however, the ambiguity of nonrigid constants is actually quite desirable. For instance, consider the sentence,

Someday, somebody will be taller than
the tallest person in the world.

A natural reading of this sentence is

The person that c designates at some future time is
taller than the person that c designates at the present time.

And this is to read the constant expression in the original sentence as non-rigid, *the tallest person in the world now*, so that it can designate different objects at different times. Of course this is not the only reading that is possible. We could read it as “At some future time the tallest person in the world at that time is taller than the tallest person at that time”, which is trivially false. Or even as “At some future time somebody who exists at the present will be taller than the tallest person at that future time.” These may not be the only possible readings, but our linguistic common sense tends to remove them from consideration. Even so, the syntactic machinery that we will introduce allows for distinct formal expressions for such things, where everyday speech is, in principle, ambiguous.

Here is another example where non-rigidity is desirable. Suppose we read the modal operator $\Box$ epistemically as “I know that.” We want to create a natural model for the situation expressed by

I know the world population is at least 5 billion,
but I do not know if it is smaller than 6 billion.

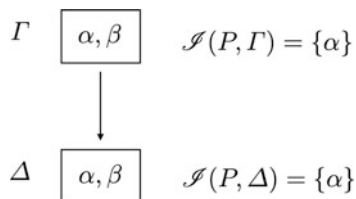
For the epistemic model we take the set of possible worlds, $\mathcal{G}$, to consist of all states of affairs that are compatible with my present knowledge. Let the constant symbol c designate the size of the world population. In the present world, c designates whatever number corresponds to the actual world population. In every possible world in $\mathcal{G}$, c designates a number that is at least 5 billion, but in some possible worlds the number that c designates is also greater than 6 billion. Then (assuming arithmetic operations have their usual interpretation) the sentence

$$\Box(c > 5 \cdot 10^9) \wedge \neg\Box(c < 6 \cdot 10^9)$$

should be true at the member of $\mathcal{G}$ corresponding to the actual state of affairs.

But the ambiguity that is introduced is a bit more complicated than just allowing constant symbols to designate different things at different worlds. We can see these complications in the following formal example.

Example 13.1.1 Let $\mathcal{M}$ be the constant domain two-world model given schematically as follows.



We have two worlds, Γ and Δ , both with domain $\{\alpha, \beta\}$, with Δ accessible from Γ . The relation symbol P is interpreted to be true of α (only) at both worlds. And let us interpret the constant symbol c non-rigidly, taking it to designate α at Γ and β at Δ .

Now, let v be an arbitrary valuation. What status should we ascribe to the following?

$$\mathcal{M}, \Gamma \Vdash_v \Diamond P(c) \tag{13.1}$$

There are *two* reasonable scenarios.

First we could say that, since we are asking about behavior at world Γ , and c designates α there, then what we could mean by (13.1) is $\mathcal{M}, \Gamma \Vdash_{v'} \Diamond P(x)$, where x is a variable and v' is like v except that $v'(x) = \alpha$. We could do this since, at Γ , c designates α . Since Δ is the only world accessible from Γ , then $\mathcal{M}, \Gamma \Vdash_{v'} \Diamond P(x)$ is equivalent to $\mathcal{M}, \Delta \Vdash_{v'} P(x)$, and this is *true*, since $v'(x) = \alpha \in \mathcal{I}(P, \Delta)$. Thus (13.1) should be true.

Second we could say that, since $\Diamond P(c)$ is a formula whose principal operation symbol is $\Diamond$, by (13.1) we could mean $\mathcal{M}, \Delta \Vdash_v P(c)$. But this in turn should be equivalent to $\mathcal{M}, \Delta \Vdash_{v'} P(x)$, where v' is like v except that $v'(x) = \beta$. This is because at Δ , c designates β . But this is *false*, since $v'(x) = \beta \notin \mathcal{I}(P, \Delta)$. Thus (13.1) should be *false*.

If a constant c designates non-rigidly, then the truth value of a sentence containing it, $\Phi(c)$, will vary depending upon how we determine the object being designated. “The tallest building in the world is in New York City” is not true now, but it has been true in the past, though not always. This is a complication we would expect from non-rigid designation. But what the example above shows is that non-rigid designation by the constant c interacts badly with the modal operator $\Diamond$ and we can get different truth values depending on whether we take designation of the constant symbol as primary, or whether we take the modal operation of moving to an alternate world as primary. And both situations come up everyday.

We have not seen all the problems yet. Consider, for instance,

The morning star is the evening star, and always will be. (13.2)

Here we mean that the *objects*, morning star and evening star, are identical and, consequently, will remain so at all future times. By contrast, in

The ancients did not know that (13.3)
the morning star was the evening star,

we mean that, while the morning star and the evening star are, in fact, identical, there are alternate situations compatible with the knowledge of the ancients in which the objects designated by “morning star” and by “evening star” are different. In (13.2), we treat designation as primary, after which we consider future states of affairs. In (13.3), we consider alternate states of affairs before we consider what is designated.

Examples of non-rigid constant symbols, like “the tallest person in the world,” are *definite descriptions*. We will not treat definite descriptions formally until Chap. 20. For the time being, terms built up from constant and function symbols will serve as a stand-in for definite descriptions. Our formal treatment in this chapter will handle constant and function symbols, so that we wind up with a full modal analog of first-order classical logic. The formal problems that arise for non-rigid constant and function symbols will also be problems for definite descriptions, and if we present an adequate treatment of these problems now, it will make things clearer when we get to Chap. 20. Our informal examples of readings for non-rigid constant

and function symbols should be taken as anticipations of the formal treatment of definite descriptions that is to come.

There is still one more important issue in connection with definite descriptions: we will be distinguishing terms that designate nonexistent objects from terms that do not designate. Assuming a temporal reading, the expression “the victorious General of Waterloo” designates a nonexistent, Wellington (he died in 1852). But the expression “the present King of France” at the present time designates no one at all. The treatment of non-designation is a separate issue from that of non-rigidity. It is more perspicuous if we deal with one problem at a time. In this chapter we show how non-rigidity can be treated formally, assuming terms always designate. In Sect. 14.5 we start dealing with non-designation. But that is near the end of this portion of our work. We begin with the basic problems raised by issues of *scope*.

Overall, the task of this chapter is set by the blunt lesson of Example 13.1.1. *Formal syntax drawn from that of classical logic cannot distinguish modal meanings we can readily distinguish intuitively.*

13.2 Scope

As we have seen, a modal sentence $\Box\Phi(c)$ containing the constant c has a syntactic ambiguity that can engender a semantic ambiguity. On the one hand, we can regard the constant as primary: c designates an object in the actual world and that object is said, in every possible world, to be Φ . On the other hand, we can regard the modal operator as primary: in every possible world, the object designated by c in that world is said to be Φ . There are, in other words, two operations involved in reading the sentence. One of these is the world-shift operation, represented by the necessity operator. This operation is familiar from earlier chapters. But, designation (of an object by a constant symbol) is also an operation. These two operations do not commute, and so there is ambiguity in reading the sentence. It is, in fact, a *scope* ambiguity.

When we add a list of numbers, e.g., $3 + 4 + 5$, the order in which we proceed does not matter. We get the same result whether we first compute $3 + 4$ and then add that result to 5, or, alternatively, whether we first compute $4 + 5$ and add that result to 3. Either way, we end up with 12.

$$\begin{array}{cc} \begin{array}{c} 12 \\ \underbrace{\quad} \\ 7 + 5 \\ \underbrace{\quad} \\ 3 + 4 + 5 \end{array} & \begin{array}{c} 12 \\ \underbrace{\quad} \\ 3 + 9 \\ \underbrace{\quad} \\ 3 + 4 + 5 \end{array} \end{array}$$

But when we mix operations, as in $3 \times 4 + 5$, ordering does matter. If we compute 3×4 first and add the result to 5, we get 17. On the other hand, if we compute $4 + 5$ first and multiply the result by 3, we get 27.

$$\begin{array}{cc} 17 & 27 \\ \underbrace{12 + 5} & \underbrace{3 \times 9} \\ \underbrace{3 \times 4 + 5} & 3 \times \underbrace{4 + 5} \end{array}$$

The notation $3 \times 4 + 5$ is therefore ambiguous: without a convention indicating the order of operations, it can denote 17 or it can denote 27. The usual way of clarifying the notation is by using parentheses: $(3 \times 4) + 5 = 17$ and $3 \times (4 + 5) = 27$.

We are familiar with this use of parentheses in propositional logic to clarify the scope of operations in a formula. For example, we distinguish $(P \vee Q) \wedge R$ from $P \vee (Q \wedge R)$. When P and Q are both true while R is false, the former comes out false while the latter comes out true. Standard notation is designed to be free of ambiguity so that a complex formula will uniquely decompose into its atoms, guaranteeing that an assignment of truth values to the atoms yields a unique truth value to the complex. But, we have reached a point where standard notation breaks down—non-rigid constant symbols make it possible to write ambiguous formal sentences.

Aristotelian logicians identify a scope ambiguity engendered by negation in a sentence like

$$\text{Wellington is } \textit{not} \text{ happy.} \quad (13.4)$$

On the one hand, it could be the sentence that is negated, i.e.,

$$\text{It is } \textit{not} \text{ the case that Wellington is happy,} \quad (13.5)$$

or it could just be the predicate that is negated, i.e.,

$$\text{Wellington is } \textit{not-happy}. \quad (13.6)$$

For the Aristotelian, (13.6) implies (13.5), but not conversely. In modern first-order logic, this distinction is not marked in the symbolism. But taking w to abbreviate “Wellington” and Hx to be the predicate “ x is happy”, (13.5) could be symbolized as $\neg(Hw)$ and (13.6) could be symbolized as $(\neg H)w$. Think of the first as denying a certain positive property to Wellington, and the second as ascribing a negative property to him. But in standard formal logic, both versions are represented by $\neg Hw$, and the Aristotelian distinction is lost. This is reasonable since in modern non-modal symbolic logic the distinction plays no role. Modally, however, the distinction is significant, and we will shortly introduce a better notation to represent it.

Distinctions like the Aristotelian one above were found to be significant when definite descriptions that lacked designations were considered, even without modal

operators present. This led to the introduction of a formal scoping mechanism by Whitehead and Russell (1925–1927) in connection with their celebrated Theory of Descriptions.¹ On their view,

The present King of France is *not* happy,

is ambiguous. Let us abbreviate “the present King of France” as f . Then, as above, we might represent the two readings by:

$$\neg(Hf), \tag{13.7}$$

and

$$(\neg H)f. \tag{13.8}$$

(This, of course, is not Russell’s notation.) In (13.7), we are denying that the present King of France has a certain property, namely happiness, whereas in (13.8), we are affirming that the present King of France has a certain property, namely non-happiness. The logical distinction between the two shows itself when we consider a situation, like now, in which there is no King of France. If we were to assume that a nonexistent being has no properties, the first should be true since it denies the King has a property, and the second should be false, since it says he has a property. But to say this clearly, we will first need a careful explanation of our scope convention.

For Russell, scope was a logical factor when the description failed to designate an existent, but even if it did designate an existent, scope was still a factor in non-truth-functional contexts. This is precisely the situation that affects us, for modal contexts are non-truth-functional.

In Russell’s terminology, in (13.7) the description f has *small* or *narrow* scope, and in (13.8), it has *large* or *wide* or *broad* scope. This is a distinction somewhat analogous to the *de re/de dicto* distinction. We will see that such distinctions are not really general enough to account for all the cases of interest, and a more versatile scoping mechanism is needed.

13.3 The De Re/De Dicto Distinction, More Examples

We introduced the distinction between *de re* and *de dicto* necessity earlier in Sect. 8.3. Here we provide more examples of the *de re/de dicto* distinction.

¹ Russell (1905) originally introduced a distinction between *primary* and *secondary* occurrences of a description to handle such modal contexts. The full *scope* treatment was not presented in any detail until Whitehead and Russell (1925–1927).

The modern paradigm of the *de re/de dicto* distinction comes from Quine (1943). He noted that

The number of planets is necessarily greater than seven (13.9)

is ambiguous. He distinguished the two readings (as best as possible in English) as follows. The *de dicto* reading is

It is necessary that
the number of planets is greater than seven (13.10)

and the *de re* reading is

The number of planets is such that
it is necessarily greater than seven. (13.11)

The *de dicto* (13.10) says the claim that the number of planets is greater than seven is necessarily true. The *de re* (13.11) says the number that numbers the planets—eight—is necessarily greater than seven. The *de dicto* reading is false: it is a contingent truth, not a necessary truth, that there are eight planets. But the *de re* reading is true. It is a necessary property of the number eight that it is greater than seven.

A medieval paradigm would involve one of the Aristotelian categorical forms. Here is an example. Contrast these two:

Necessarily every bachelor is unmarried;
Every bachelor is necessarily unmarried.

In modern logical notation, the distinction is (with ‘Fx’ abbreviating *x is a bachelor* and ‘Gx’ abbreviating *x is unmarried*),

$$\begin{aligned} &\Box(\forall x)(Fx \supset Gx) \\ &(\forall x)(Fx \supset \Box Gx) \end{aligned}$$

In each case, the first is the *de dicto* form and the second is the *de re* form. Note that the *de dicto* form is plausibly true: expanding “bachelor” to its definitional equivalent “unmarried male”, it turns out to be a logical truth, and therefore a necessary truth. But the *de re* form is false: surely we can find a bachelor who, in another possible world, is married. It need not be an essential property of him that he is not married.

Here is one more important example using the alethic modality. There are two ways of formally representing the sentence “Something is necessarily *F*,” namely,

de dicto: $\Box \exists x Fx$ Necessarily something is *F*

de re: $\exists x \Box Fx$ Something is necessarily *F*.

This is a classic way of drawing the distinction. The *de re* case—but not the *de dicto* case—involves our “quantifying in” to a modal context: a quantifier lying outside the scope of a modal operator binds a variable that lies inside its scope.

The distinction is a logical one, not a metaphysical one, and it is repeated across the modals. Here it appears in a temporal framework:

At some future time, the Prime Minister will be a Laborite (13.12)

The Prime Minister will at some future time be a Laborite. (13.13)

Here (13.12) is the *de dicto* form. It says, in effect, that there is a time in the future at which a Laborite will be the Prime Minister. (13.13) is the *de re* form. It says that the current Prime Minister will, at some time in the future, be a Laborite. This is a remarkable claim if the current Prime Minister is actually a Conservative: (13.13) predicts he will change his political affiliation.

Here it appears in an epistemic framework. Reflect on the ambiguity of

John knows that the President of the United States

is at least 35 years of age. (13.14)

John might have knowledge of the requirements for President set forth in the U. S. Constitution: it states that the President must be at least 35 years old. To interpret (13.14) this way is to assign it a *de dicto* reading. On the *de re* reading, we understand John to know *of* the President—of that man (John might not even know he is the President)—that he is at least 35 years old.

We find it in statements of desire. Here is another of Quine’s examples,

John wants a sloop.

There might be a particular sloop John wants. This is the *de re* reading. Or John might seek relief from—to use Quine’s neologism—“slooplessness.” This is the *de dicto* reading.

Finally, we note its appearance in a doxastic framework. The example is actually from Russell. He noted the ambiguity of

I thought your yacht was longer than it is. (13.15)

On the *de dicto* reading, your thought is contradictory: you thought the length of your yacht was greater than the length of your yacht. It is the *de re* reading that would most charitably be understood here: the actual length of your yacht is greater than the length I thought it had.

13.4 The De Re/De Dicto Distinction: History

Although the *de re/de dicto* terminology was not introduced until the thirteenth century (by St. Thomas Aquinas), the distinction, under various names, had been a feature of modal logic in the West (i.e., in Europe) as early as the beginning of the twelfth century. It was then that Peter Abélard established a central place for a *de rebus/de sensu* distinction in modal reasoning. Abélard used what the medievals called *formal* and *personal* Latin forms to conventionally contrast the (false) *de sensu* claim,

It is possible *that* the one who is not writing is writing,

from the (true) *de rebus* claim,

It is possible *for* the one who is not writing to be writing.

To be sure, one cannot write and not write at the same time, but one can write at one time and not write at another.

Abélard claimed to be making the very same distinction as that drawn by Aristotle in *Sophistical Refutations*, I, 166a22-30, under the *Fallacies of Composition and Division*.

But the following are referred to composition: as, that is is possible for him who sits to walk and for him who does not write, to write. For it does not signify the same thing to say separately and conjointly, that he who sits may walk, and that he who does not write, may write; and in a similar manner this, if some one should conjoin [the words], that he who does not write, writes. For it signifies that he has a power by not writing, of writing. But, if the words are not conjoined, it signifies, that he has a power when he does not write, of writing. (Aristotle 2002, p. 637)

Aristotle's authorship of the distinction was embedded in the medieval literature where the two readings were commonly designated *necessity in the divided sense* and *necessity in the combined sense*.² Primarily von Wright (1951) and Prior (1952) to a lesser degree, are responsible for reintroducing this to a modern audience.

The relation between the two forms was controversial from the very beginning of the medieval period. Abélard opposed "the teaching of the master"—widely thought

² For the history of modality in medieval logic, see Knuuttila (1993), Thom (2003), and Lagerlund (2016).

to be William of Champeaux—that all necessity was *de sensu*. Abélard developed a genuine necessity *de rebus*, and having done so, took a step further: he went on to hold that all necessity, including William’s necessity *de sensu*, was ultimately *de rebus*. Abélard’s conception fits comfortably with modern thinking. Kripke-style possible world semantics for first-order modal logic is built on an object-grounded conception of modality, by contrast with the Intensional Logic of Church (1951), which is built on Frege’s concept of Sense (*Sinn*), Frege (1982b). Abélard had only limited access to Aristotle’s work. The full corpus, including Aristotle’s treatment of the modal syllogism in *Prior Analytics* was not translated into Latin until much later in the twelfth century.

The situation was far different in the East. Aristotle’s works were fully translated into Arabic as early as the ninth century. Al-Farabi had access to them, but no commentaries by him on the modal syllogism have survived. However, Avicenna, writing a century after Al-Farabi (and a century before Abélard), produced a number of commentaries on Aristotle’s modal syllogistic. Nonetheless, as Tony Street, reports: “There is no distinction made in Arabic logic corresponding to the Western distinction between divided and composite readings,” (Street, 2004, 550). Other distinctions due to Avicenna became of central importance in the Eastern tradition:

Necessity may be (1) absolute, as in *God exists*; or it may be connected to a condition. The condition may be (2) perpetual for the existence of the substance, as in *man is necessarily a rational body*. By this we do not mean to say that man has been and always will be a rational body, because that would be false for each given man. Rather, we mean to assert that while he exists as a substance, as a human, he is a rational body. Or the condition may be (3) perpetual for the subject’s being described in the way it is, as in all mobile things are changing; this is not to be taken to assert that this is the case absolutely, nor for the time [the subject] exists as a substance, but rather while the substance of the moving thing is moving. Distinguish between this condition and the first condition, because the first has set down as the condition the principle of the substance, ‘man’, whereas here the substance is set down with a description that attaches to the substance, ‘moving thing’. ‘Moving thing’ involves a substance to which movement and non-movement attach, but ‘man’ and ‘black’ are not like that. (Street, 2004, 551).

With reference to items (1), (2), and (3) in the quote above, Street further remarks that, although different from the divided/composite distinction, the conditional necessities (2) and (3) make for a distinction that is “functionally the same.” He gives two examples:

All bachelors are necessarily unmarried is true as a [type (3) conditional necessity], because ‘bachelors’ picks out men just while they are unmarried: *all men while bachelors are necessarily unmarried*. As a [type (2) conditional necessity], however, it is false: all bachelors are men, and it is untrue that all men are necessarily unmarried. By contrast, (and this is the most common Avicennan example) *all who sleep wake* is true as a [type (2) conditional necessity] (because every animal that sleeps also wakes up from time to time), but false as a [type (3) conditional necessity] (because nothing can be awake while sleeping). (Street (2004), p. 552)

Avicenna’s treatment of modal logic is rich and inventive. For a close study of Avicenna’s work on expressions of the sort “All things that are at all times F

are necessarily at some times G ", where we have quantification over objects and quantification over times, see the largely unpublished Hodges (2022).

But whether Aristotle himself appreciated the *de re/de dicto* distinction is far from clear. His account of the modal syllogism is confused. Some of what he says is consistent with a *de re* reading; some with a *de dicto* reading. There are scholars who claim that neither of these capture Aristotle's modal intuitions. The quotation from Abélard's *Sophistical Refutations* near the beginning of this section does not settle the issue. After all, the *combined/divided* difference might pertain only to the predicate, so that both readings turn out to be *de re*: 'The individual is possibly both walking and not walking' vs 'The individual is walking but possibly not walking'.³

13.5 Understanding the Distinction: Possible Worlds and Scope

Here is how St. Thomas characterized the distinction:

A modal proposition is either *de dicto* or *de re*. A modal proposition *de dicto* is one in which the whole *dictum* is the subject and the mode is the predicate, as when it is said 'For Socrates to run is possible'. A modal proposition *de re* is one where the mode is interpolated in the *dictum*, as when it is said 'For Socrates it is possible to run'. Aquinas (1976), p. 421: 42–47⁴

A *de dicto* necessity is a claim about the necessity of a *dictum*, what we think of today as a *proposition*, and a *de re* necessity is a claim about the necessity of a *res*, a thing or an object. However, in the idiom of possible world semantics the two readings appear to be about objects. For the *de dicto* reading, $\Box(Fa)$, we move to a possible world, pick out in that world the object that is a , and evaluate in that world whether it is F . The designator a lies within the scope of the modal operator so its semantic designation is determined by its interpretation in the possible world. For the *de re* reading, $(\Box F)a$, we pick out in this world the object that is a and then move with it to the possible world and evaluate in that world whether it is F . The designator a lies outside the scope of the modal operator so its semantic interpretation is fixed in the actual world. For the *de dicto* reading, we move and

³ Letting ' Wx ' abbreviate x is walking, the distinction looks like this, using predicate abstract notation which will be introduced properly starting in Sect. 13.6:

$$\begin{aligned} &(\exists x)(\lambda x.\Diamond(W \wedge \neg W)x)(x) \\ &(\exists x)(\lambda x.(W \wedge \Diamond\neg W)x)(x) \end{aligned}$$

The first is the combined case: the conjunction of the predicates is said to be possible; the second is the divided case: only the second of the conjoined predicates is said to be possible. In both cases, the modal governs an open sentence, so both are examples of *de re* possibility.

⁴ The translation is by Novaes (2003).

then designate; for the *de re* reading, we designate and then move. What is critical is the order of these operations, and that is how the matter of scope has become salient.

In the heuristics of possible world semantics, $\Box$ and $\Diamond$ are regarded as quantifiers over possible worlds. $\Box P$ says *P holds in every possible world*: perhaps, $\forall i P$ in i .⁵ $\Diamond P$ says *P holds in some possible world*: $\exists i P$ in i . Similarly for the first-order case: $\Box Fa$ says *Fa holds in every possible world*; and similarly for $\Diamond Fa$. When we use a quantifier $\forall$, we must specify the bound variable: in this case, it is the index that relativizes the relevant syntactic parts to a world. So, the natural reading of $\Box Fa$ is

$$\text{In every possible world } w_i, a \text{ in } w_i \text{ is } F \text{ in } w_i. \quad (13.16)$$

It is helpful on occasion to speak of possible worlds as interpretations for the linguistic expressions; so we can read (13.16) as

$$\text{Under every semantic interpretation } w_i, a \text{ in } w_i \text{ is } F \text{ in } w_i, \quad (13.17)$$

Note that we have assigned both parts of the sentence—subject and predicate—the same interpretation. This is the most natural understanding; because when we apply an interpretation to a sentence Fa we suppose that all parts of the sentence are being evaluated under the same interpretation. Equation (13.17) is the *de dicto* reading. But, we might also have the following situation: one part of the expression is to be evaluated on one interpretation w_i and another part of the expression is to be evaluated on another interpretation w_j .

$$a \text{ in } w_i \text{ is } F \text{ under every interpretation } j. \quad (13.18)$$

This is the typical situation we face in the case of a *de re* modal claim: the subject is governed by one interpretation w_i , and the predicate is governed by another interpretation w_j . In the *de dicto* sense—the combined sense—both parts of the sentence signify under the same interpretation. In the *de re* sense—the divided sense—they do not: the subject picked out under the first interpretation serves as the argument for which the predicate is evaluated under the second interpretation. Note, by the way, that we could have unambiguously expressed (13.18) by

$$\text{In every possible world } w_j, a \text{ in } w_i \text{ is } F \text{ in } w_j. \quad (13.19)$$

In (13.19), the modal is out front governing the whole sentence, so one might think that (13.19) is a *de dicto* claim. But the subject-expression in (13.19)— a in w_i —despite being located inside the scope of that modal operator, is not bound by it: the associated indices make that clear. (13.19) is actually *de re*. There is a moral to be drawn here. The usual rules of thumb for determining *de dicto* as opposed to

⁵ We should more carefully say “every possible *accessible* world.” But we will set aside such precision so that the intuitive reading in this informal discussion is unencumbered.

de re constructions can mislead, and one must be careful that the logical roles are correctly identified.

This notion we have just been describing is the *de re* notion that is captured by the following modern definition given by Kit Fine:

A formula φ is *de dicto* if no free variable occurs within the scope of $\Box$. In other words, in a *de dicto* formula each subformula of the form $\Box\psi$ is also a sentence. A formula is *de re* if it is not *de dicto*. Fine (1978), p. 135.

Fine's distinction relies very much on the positioning of syntactical elements. Again, that's how scope is involved. Note that in (13.19), w_i is a free variable. In that context it is regarded as a *deictic* pronoun for the actual world. Some prefer a name for the actual world—commonly $w_@$ —and represent the *de re* case like this:

$$\text{In every possible world } w_j, a \text{ in } w_@ \text{ is } F \text{ in } w_j, \quad (13.20)$$

But Fine's characterization is more general. If, in any subformula, we have a variable that lies outside the scope of a governing modal, we have a *de re* construction. So, for example, consider the formula

$$\Box\forall x(Fx \supset \Diamond\exists yRxy). \quad (13.21)$$

The variable x inside the consequent $\Diamond\exists yRxy$ —which is, of course, a subformula—is bound by a quantifier that lies outside the scope of the $\Diamond$ operator. But in the subformula itself it is free. So even though we have a modal out front governing the entire sentence, (13.21) is not *de dicto* on Fine's definition; it is *de re*. Now that free variable in the consequent does not range over objects in this, the actual world; it is controlled by the $\Box$ operator in the front of the formula and so it ranges over objects in the possible world under consideration in the antecedent. So, the modern notion of *de re* is more general than the original. It captures the idea of moving with an object from one world to another, not just from *this* world to another.

This reading of the *de re* reveals its connection to one of the main strands of what has come to be known as *Two Dimensional Semantics*. Two Dimensional Semantics found its origin in the double indexing introduced to formalize temporal reasoning (Vlach, 1973; Kamp, 1971). When we analyze a sentence like

$$\begin{array}{l} \text{At some time in the future,} \\ \text{this infant prince will ascend to the throne,} \end{array} \quad (13.22)$$

there is the time of utterance of the statement, and it is at this time that the infant prince is picked out; and there is the time in the future when he ascends the throne, a time, perhaps, when he is no longer an infant. We have the temporal world in which the infant prince is identified and we have the temporal world in which the crown is put on his head. The reader can recognize (13.22) as a *de re* claim. The two indices appear in David Kaplan's analysis of indexicals as the *context of utterance* and *circumstance of evaluation* (Kaplan, 1989). A sentence containing

an indexical receives two distinct interpretations. The sentence is not ambiguous. Rather, it goes through a two-step interpretational process: the indexical is first interpreted in a context of utterance. The result of that interpretation is then piped into the interpretation for the predicate which is evaluated in the circumstance of evaluation. Kaplan then takes the proposition expressed to be an ordered pair whose first element is the object picked out in the context of utterance and whose second element is the property associated with the predicate that is picked out in the circumstance of evaluation. This was sometimes termed *a singular proposition*. But there are other forms of Two Dimensional Semantics, some of which are much more philosophically contentious. For a sampling, see the essays contributed by some of the leading theorists to a collection in the journal *Philosophical Studies* (Davies and Stoljar, 2004).

Exercises

Exercise 13.5.1 Give an example illustrating the *de re/de dicto* distinction for the epistemic case. Similarly for the temporal case.

13.6 Predicate Abstraction: Informal Discussion

We sometimes think of classical first-order formulas as representing *predicates*. A formula such as $\Phi(x)$, with one free variable, defines in each model a certain class of objects—the members of the domain of that model for which $\Phi(x)$ is true. There is a harmless ambiguity in the notation that generally goes unnoticed classically. Suppose $\Phi(x)$ is the formula $P(x) \wedge Q(x)$. We take this formula to express the conjunction of two simpler predicates, P and Q ; but, alternatively, we could have taken it to express the application of a compound predicate $[P \wedge Q]$. We will not try to make the distinction any clearer since, under any plausible interpretation, the results will be the same.

In a modal setting, however, things are quite different. We saw in Example 13.1.1 that $\Diamond P(c)$ gives different results if we think of it as the “possible- P ” predicate applied to c , or if we think of it as asserting that the P predicate applied to c is possible. Formulas can no longer be thought of as representing predicates, pure and simple. Rather, a representation of a predicate can be *abstracted* from a formula. This is the purpose of the device of predicate abstraction. We make a notational distinction between a formula,

$$\Phi(x),$$

and the predicate abstracted from it,

$$\langle \lambda x. \Phi(x) \rangle.$$

This is by analogy with the *lambda-calculus*, in which a distinction is made between an *expression* like $x + 3$, and the *function abstracted from it*, $\langle \lambda x. x + 3 \rangle$. Once the formal machinery is in place, starting in Chap. 14, we will see, as promised, that

$$\langle \lambda x. (P(x) \wedge Q(x)) \rangle(c)$$

and

$$\langle \lambda x. P(x) \rangle(c) \wedge \langle \lambda x. Q(x) \rangle(c)$$

do indeed behave alike. On the other hand,

$$\langle \lambda x. \Diamond P(x) \rangle(c)$$

and

$$\Diamond \langle \lambda x. P(x) \rangle(c)$$

are quite different, with behaviors corresponding to the two readings that arose in Example 13.1.1.

We have been abbreviating formulas like $(P(x) \wedge Q(x))$ by $P(x) \wedge Q(x)$, omitting outer parentheses. We will often apply a similar device within predicate abstracts, abbreviating $\langle \lambda x. (P(x) \wedge Q(x)) \rangle(c)$ by $\langle \lambda x. P(x) \wedge Q(x) \rangle(c)$. No ambiguity arises by doing this, and since parentheses multiply when predicate abstraction is involved, clarity is often enhanced by parenthesis omission.

13.7 The Scope Distinction and Predicate Abstraction: Informal Discussion

The scope distinction, as the generalized *de dicto/de re* distinction has come to be known—has roots deep in the history of philosophy, but its name and application to modal contexts is relatively recent. Whitehead and Russell (1925–1927) created a notation that provided a user interface in which descriptions appear as constants on top of an underlying code in which descriptions are treated as quantifier expressions. Arthur Smullyan (1948) borrowed their somewhat cumbersome notation for delineating the scope of a definite description in a complex sentence, and applied it to disambiguate the result of placing a modal operator in front of a sentence constructed in classical first-order syntax.

The precursor of Russell and Whitehead's more general scope distinction was Russell's distinction between the *primary* and the *secondary* occurrence of a term, which he introduced in Russell (1905) to handle the ambiguity in a sentence like $\neg Fa$ that arises when we allow for the possibility that a does not designate anything. So long as we require that a constant denote, there is no ambiguity. But 'the largest natural number' does not designate anything: for any natural number you identify, there is another one that is larger. If we take the claim

The largest natural number is not odd (13.23)

to be what Russell called "a subject-predicate sentence," i.e., a sentence of the form Fa , then we regard the subject 'the largest natural number' as designating something and we regard the predicate 'is not odd' as saying of or about that something that it lacks the property of oddness. But 'the largest natural number' does not designate anything. There is, therefore, nothing to say anything about. It looks as though (13.23) is neither true nor false. Another way of making this point, closer to the way Russell (1905) made it, would be this: If we look up all the numbers that are odd, we will not find anything that counts as the largest natural number; and if we look up all the numbers that are not odd, we will not find anything that counts as the largest natural number. Russell suggested that we read (13.23) as the negation of the claim that the largest natural number is odd. Taken this way, (13.23) looks to be true, and, moreover, it does not imply that the largest natural number is not odd. So, Russell distinguished these two by pointing to the scope of $\neg$ in $\neg Fa$. It might be read as $(\neg F)a$, where the negation sign combines with the predicate F to form a complex predicate $\neg F$ that is predicated of a , or as $\neg(Fa)$, where the negation sign governs the whole sentence.

Thomason and Stalnaker (1968)⁶ introduced a notational device for predicate abstraction; they used Russell's circumflex notation $\hat{x}$, while we use the lambda notation λx from Church (1932). Predicate abstraction works very much like λ -abstraction in the lambda calculus, but without what is called the λ -conversion rule, which says $\langle \lambda x.Fx \rangle(a) \equiv Fa$. The λ -conversion rule would collapse the *de re/de dicto* distinction: both $\Box \langle \lambda x.Fx \rangle(a)$ and $\langle \lambda x.\Box Fx \rangle(a)$ would turn out to be equivalent to $\Box Fa$.

Predicate abstract notation perspicuously represents the binding character of anaphora: λ collects together the various places in the predicate where the same term is to be inserted and the symbol in parentheses designates the value piped through. It is this latter operation that we refer to as 'binding'. The common notation for First-Order Logic conflates these two uses of variables. Quine (1950) made considerable effort to distinguish them, using circled numerals in the first case and lower case Roman letters from the back of the alphabet for the second. Frege (1892a) emphasized a similar distinction. He used Greek letters to fill the first role but he did

⁶ See also Stalnaker and Thomason (1968).

not use variables for the second role, but only constants. Frege did not admit of free variables.

Here is the translation from the usual first-order notation to our predicate abstract notation:

$$Fa \rightsquigarrow \langle \lambda x. Fx \rangle(a) : a(\text{has the property of being } F).$$

The constant binds the lambda's variable just as a quantifier does. Semantically, of course, the two differ in that the individual constant designates a unique object while the existential quantifier, supposing it to designate, designates a disjunction of them. The device effectively permits us to transform complex sentences of first-order logic into complex predicates.

$$\begin{aligned} Fa \wedge Ga &\rightsquigarrow \langle \lambda x. (Fx \wedge Gx) \rangle(a) : a \text{ has the property of being } F \text{ and } G \\ Fa \supset Ga &\rightsquigarrow \langle \lambda x. (Fx \supset Gx) \rangle(a) : a \text{ has the property of being if } F \text{ then } G \end{aligned}$$

This ability to form complex predicates enables us to formally represent scope issues. Russell and Whitehead's scope distinction for sentences involving negation is represented in predicate abstract notation as follows:

$$\neg Fa \rightsquigarrow \begin{cases} (i) & \langle \lambda x. \neg Fx \rangle(a); \\ (ii) & \neg \langle \lambda x. Fx \rangle(a). \end{cases}$$

(1) says that a has the $\neg F$ property; (2) says that it is not the case that a has the F property. When a is, in Russell's words, a logically proper name, scope makes no difference.⁷

Informal Definition a is a *Logically Proper Name* if $\langle \lambda x. \neg Fx \rangle(a) \equiv \neg \langle \lambda x. Fx \rangle(a)$.

On the usual rendering of first-order logic, constants are subject to both Universal Instantiation and Existential Generalization, and there is no logical difference corresponding to the typographical difference between Φa and $\langle \lambda x. \Phi x \rangle(a)$. But if we are in a first-order logic where designation is impacted by existence issues, the distinction is relevant.

Scope issues are of paramount importance in first-order *modal* logic. The ambiguous $\Box Fa$ becomes in predicate abstract notation one of the following unambiguous forms:

$$\Box Fa \rightsquigarrow \begin{cases} (i) & \langle \lambda x. \Box Fx \rangle(a); \\ (ii) & \Box \langle \lambda x. Fx \rangle(a). \end{cases}$$

⁷ “[A]n expression N is being used as a Russellian name only if, where δ is any expression that forms sentences from sentences and predicates from predicates, e.g., ‘not’, ‘allegedly’, or ‘possibly’, ‘ $\lceil \delta \phi \rceil N$ ’ is indistinguishable in sense from ‘ $\lceil \delta(\phi N) \rceil$ ’”, (Prior, 1971, 150).

(1) is the *de re* reading, the box attaches to the predicate only: $\langle \lambda x. \Box Fx \rangle(a)$ says *a has the property of being necessarily F*. (2) is the *de dicto* reading, the box governs the whole statement: $\Box \langle \lambda x. Fx \rangle(a)$ says *it is necessary that a has the F property*. (Similarly for $\Diamond Fa$.) Predicate abstract notation makes clear that, in the *de re* case, the modal operator governs an otherwise free position that is bound by an expression that resides outside the scope of the modal operator.

In the case that scope does not matter, our constant is *rigid*.

Informal Definition *a* is a *Rigid Designator* if $\langle \lambda x. \Box Fx \rangle(a) \equiv \Box \langle \lambda x. Fx \rangle(a)$.

Rigidity is the modal analog of Russell's *logically proper name* in first-order logic. The Informal Definition above is not actually what we will take as our definition of rigid designator, but it does serve to some degree as motivational. The full situation is more complicated, and will be discussed at length in Chap. 19.

13.8 Reading Predicate Abstracts

Consider the following example. Suppose the date is January 1, 1902. The current President of the United States, Theodore Roosevelt, is 43. Now, the sentence,

George knows the President is at least 35, (13.24)

has two interpretations. First, George might know that Roosevelt is at least 35. On this reading, (13.24) implicitly binds the term “President” to the person Theodore Roosevelt so that it is taken to be an assertion specifically about *him*. (George might have known Theodore Roosevelt from childhood, and know he is at least 35 without even knowing he is President.) Second, George might have in mind the requirement in the U.S. constitution that a President be at least 35. In this case George is not specifically referring to Theodore Roosevelt—indeed, he might not even know who the current President is. Formalizing these two versions involves different bindings.

Take $\Box$ to be “George knows”; take $A(x)$ to be “*x* is at least 35”; and take p to be a non-rigid constant symbol “President of the United States.” For the formula

$$\langle \lambda x. \Box A(x) \rangle(p)$$

to be true at January 1, 1902, the person designated by p must satisfy the condition $\Box A(x)$. Rather awkwardly then, we could read the formal sentence as, “It is true of the person who is now the President of the United States, that George knows the person’s age is at least 35.” The more colloquial reading is, “George knows, of the current President, that he is at least 35.” Placing $\Box$ inside a predicate abstract generally can be read using an “of” phrasing.

On the other hand, the formula

$$\Box \langle \lambda x. A(x) \rangle(p)$$

can be read, mechanically, as “George knows that whoever is the President of the United States, the person is at least 35.” The more colloquial reading is “George knows that the President is at least 35.” This time we used a “that” reading.

Here is another example. One of the authors of this book once had a dog named Benedict who, late in life, had a leg amputated. Suppose you, a stranger to the dog, saw him. You directly see a dog with three legs, and so you know that the dog you are looking at has three legs. That dog is Benedict, and so you know, of the dog Benedict, that he has three legs. But suppose you were never told the name of the dog. Then while you know, *of* Benedict, that he has three legs, you don’t know the truth of the assertion *that* Benedict has three legs. It is true that Benedict has three legs, but you don’t know that.

This is a distinction we make every day, but generally it is implicit in our conversation and phrasing. It is part of the unspoken background. When we must put it into words, the *of*/*that* phrasing used above is probably the best we can do. The reason it seems a little awkward is simply that so often we never do put it explicitly into words—we rely on context to clarify. Nonetheless, when formalizing, such distinctions must appear.

In this text we have adopted the notational device of *predicate abstraction*, which will enable us to make the relevant distinctions we have discussed, including the *de re/de dicto* distinction, readily apparent as scope distinctions.

As we have seen in discussing Russell’s scope distinction, we identified an ambiguity in the English sentence

The King of France is not happy

that depended upon whether we regarded the “not” as operating on the whole sentence or just on the predicate. Using predicate abstract notation, we can symbolize the two readings as follows:

$$\neg\langle\lambda x.H(x)\rangle(f) \tag{13.25}$$

and

$$\langle\lambda x.\neg H(x)\rangle(f). \tag{13.26}$$

(13.25) denies *that* the claim Hf is true; (13.26) says *of* f , that *he is not* H . (Compare the versions above, using predicate abstraction, with the unofficial versions (13.7) and (13.8).)

Predicate abstraction notation allows us to make a similar distinction for the temporal examples (13.12) and (13.13) from Sect. 13.2. We distinguish

$$\mathbf{F}\langle\lambda x.L(x)\rangle(m) \tag{13.27}$$

from

$$\langle \lambda x. \mathbf{FL}(x) \rangle (m) \quad (13.28)$$

In the first, we attribute *being a Laborite* to a future Prime Minister; in the second, we apply the property of *being a future Laborite* to the present Prime Minister.

Predicate abstraction also disambiguates another problem which might arise when we have a relational expression which can be analyzed in different ways. Consider the following example.

$$\text{Caesar killed Caesar.} \quad (13.29)$$

This can be analyzed in two ways:

- The two-place relation “ x killed y ” is applied to the ordered pair $\langle \text{Caesar}, \text{Caesar} \rangle$: $\langle \lambda x. \langle \lambda y. x \text{ killed } y \rangle (\text{Caesar}) \rangle (\text{Caesar})$.
- The one-place predicate “ x killed x ” is applied to Caesar: $\langle \lambda x. x \text{ killed } x \rangle (\text{Caesar})$.

These represent the two ways in which a predicate can be abstracted from (13.29). In the latter case we have what is known as a reflexive predicate. In the former case, we do not. The distinction is lost when we see the final sentence, “Caesar killed Caesar.”

Exercises

Exercise 13.8.1 Predicate abstraction can be tricky when we have more than one variable involved. Consider the sentence, “Necessarily Tolstoy authored *War and Peace*.” Using predicate abstraction and the necessity operator, $\Box$, formalize this sentence in as many ways as you can. Then explain whether the different formalizations express different readings or not.

13.9 Actuality

The actuality operator $\mathbb{A}$ that Crossley and Humberstone (1977) introduced into the modern logical toolkit has historical antecedents stretching deep into the past. The great fourteenth century logician, Buridan (2015), had the notion of *ut nunc* necessity—as of now or for now—that served a similar purpose. On Buridan’s view, in a sentence of the form *A is necessarily B*, the supposition of the subject ampliates

to include not just the contemporary but also the possible. To borrow a famous example, Buridan would say that in

The number of planets is necessarily greater than seven, (13.30)

we are speaking not only of the number that in fact numbers the planets but the possible numbers that number the planets, and what (13.30) says is that every one of these is greater than seven. Clearly, (13.30), so understood, is false. But in the *ut nunc* form, for which his English translator used the idiom “that which is,” viz.

That number which is the number of planets
is necessarily greater than seven. (13.31)

the supposition of the subject does not amplify, but is instead restricted to that number which is the number of planets: (13.31) is true, because that number which is the number of planets, namely eight, is such that it is necessarily greater than seven.

Equation (13.30) is what we would call the *de dicto* necessity, and (13.31), the *de re* necessity. Buridan provided a stylized language in which one is able to unambiguously express these two forms. There is no syntactic maneuvering of the *necessarily* to bring the subject into its scope—or, alternatively, to keep it out. No, the syntax remains the same, and the *ut nunc* particle does all the work. Buridan’s form (13.31) is very much like the modern

The number which *in fact*—or *actually*—numbers the planets
is necessarily greater than seven. (13.32)

which philosophers have used to express the *de re* claim.

Crossley and Humberstone (1977) motivated the actuality operator as a device that enables us to formalize the following statement.

It is possible for every red thing to be shiny. (13.33)

Their idea was that (13.33) was to be expressed as

It is possible that all the objects that are actually red
should be shiny; (13.34)

and symbolized as follows:

$\Diamond(\forall x)(\mathbb{A}Rx \supset Sx).$ (13.35)

Actually we will show in Sect. 14.8 how to represent (13.33) using a scope apparatus, but it requires an extension of our basic machinery. You may find that the actuality operator provides a more natural representation.

But now we have to explain informally the semantics of the actuality operator.

13.10 What Is the Actuality Operator?

Here is how we understand the actuality operator:

$$\mathbb{A}p \text{ is true iff } p \text{ is true in } w_@. \quad (13.36)$$

Here is its truth clause (where $\mathcal{M}$ is our model, w_i a possible world in the model, and $w_@$ the *actual* world):

$$\mathcal{M}, w_i \Vdash \mathbb{A}p \text{ iff } \mathcal{M}, w_@ \Vdash p. \quad (13.37)$$

Here is the truth clause presented more in line with the usual modal truth clause. Let $w_i \mathcal{R} w_j$ be our accessibility relation with $w_j = w_@$. This ensures that $w_@$ is the only world accessible to w_i . Then,

$$\mathcal{M}, w_i \Vdash \mathbb{A}p \text{ iff } w_i \mathcal{R} w_@ \text{ and } \mathcal{M}, w_@ \Vdash p. \quad (13.38)$$

The actuality operator is a modal operator. We can recognize it most clearly when we turn to possible worlds. $\Box$ is a universal quantifier over possible worlds—for all worlds w_i —and $\Diamond$ is an existential quantifier over possible worlds—for some world w_i . So, with these two operators, we can speak about what holds for *all* possible worlds or, again, about what holds for *some* possible world. The actuality operator $\mathbb{A}$, by contrast, enables us to speak about a particular world, the actual world. The study of these various terms for individual possible worlds is *Hybrid Logic* (see the discussion of hybrid tableaux in Sect. 7.7). The interest in hybrid logics is clearest in a temporal framework. Where the modal equivalents in temporal logic would be *At all future times/At some future time*, hybrid logic enables the study of such expressions as *in two days time* or *two days ago*. Origins of hybrid logic can be found in Prior (1957, 1967, 1968), and modern treatments are in Blackburn et al. (2001, Section 7.3) and Areces and ten Cate (2007).

The actuality operator has some interesting logical interactions. First, iterations don't matter, i.e.,

$$\underbrace{\mathbb{A}\mathbb{A}\dots\mathbb{A}}_n p \equiv \mathbb{A}p. \quad (13.39)$$

But also, this is true

$$p \equiv \mathbb{A}p. \quad (13.40)$$

Nonetheless,

$$\Box p \not\equiv \Box \mathbb{A}p. \quad (13.41)$$

This follows immediately from the following two facts. First, we know that (13.42) is false.

$$p \supset \Box p \quad (13.42)$$

On the other hand, we know that (13.43) is true:

$$\mathbb{A}p \supset \Box \mathbb{A}p \quad (13.43)$$

The consequent of (13.43) is false if, at any given world, p is not true in $w@$. But that cannot be if the antecedent of (13.43) is true.

Medieval logicians identified and named the following two forms:

$$\text{Necessitas Consequentiae: } \Box(p \supset q) \quad (13.44)$$

$$\text{Necessitas Consequentis: } p \supset \Box q \quad (13.45)$$

There are common expressions in natural language that prove to be confusing over whether they express *the necessity of a consequence* or *the necessity of a consequent*, for example, ‘If p then necessarily q .’

The contrast is particularly vivid when we replace q by p . The *Necessitas Consequentiae* form

$$\Box(p \supset p) \quad (13.46)$$

is clearly true: $p \supset p$ is a logical truth and therefore necessary. The *Necessitas Consequentis* form, which we’ve seen already,

$$p \supset \Box p, \quad (13.47)$$

is a notorious fallacy: p can be true without being necessary. Now, with (13.46) and (13.47), we have, not only what appears to be a combined and divided modal, but in (13.47) we have a propositional letter occurring both inside and outside the scope of a box, and this makes it look *de re*. But despite the common expression “propositional variable,” p and q are not commonly regarded as variables, nor indeed

as singular terms that fill variable positions and stand for objects. On the other hand, if we were to regard propositional letters as quantifiable variables, then

$$(\forall p)(p \supset \Box p) \quad (13.48)$$

would be *de re*.

One final point. If we think of a propositional letter p as a designator for a truth value, then it is a *nonrigid designator*: an atomic letter, being contingent, will designate different truth values in different possible worlds. But when we put an actuality operator in front of it to form $\Box p$, we have a *rigid designator*: it designates the same truth value in every possible world. The actuality operator is what is known in the literature as a *rigidifier*.

References

- Aquinas, S. T. (1976). De Propositionibus Modalibus—. In *Opera omnia* (Vol. Tomus XLIII). San Tommaso.
- Areces, C. & ten Cate, B. (2007). In P. Blackburn, J. V. Benthem, & F. Wolter (Eds.), *Handbook of modal logics* (Chap. 14: Hybrid Logics, pp. 821–868). Elsevier.
- Aristotle. (2002). In T. Taylor (Ed.), *The works of Aristotle*. Prometheus Text.
- Beaney, M. (Ed.). (1997). *The Frege reader*. Blackwell.
- Blackburn, P., de Rijke, M., & Venema, Y. (2001). *Modal logic*. Tracts in Theoretical Computer Science. Cambridge University Press.
- Blackburn, P., Benthem, J. V., & Wolter, F. (Eds.). (2007). *Handbook of Modal Logic*. Studies in Logic and Practical Reasoning. Elsevier.
- Buridan, J. (2015). *Treatise on Consequences*. Fordham University Press. S. Read, Trans.
- Church, A. (1932). A set of postulates for the foundation of logic. *Annals of Mathematics*, 33, 346–366.
- Church, A. (1951). A formulation of the logic of sense and denotation. In H. M. K. P. Henle & S. K. Langer (Eds.), *Structure, method, and meaning: essays in honor of Henry M. Sheffer*. Liberal Arts Press.
- Crossley, J. N. & Humberstone, I. L. (1977). The logic of “actually”. *Reports on Mathematical Logic*, 8, 11–29.
- Davies, M. & Stoljar, D. (Eds.). (2004). The two-dimensional framework and its applications: metaphysics, language, and mind. *Philosophical studies* (Vol. 118, 1, 2, 3). Springer
- Fine, K. (1978). Model theory for modal logic part I: The *de re/de dicto* distinction. *The Journal of Philosophical Logic*, 7, 125–156.
- Fitting, M. (1996). *First-order logic and automated theorem proving* (1st ed.). Springer-Verlag. 1990.
- Frege, G. (1892a). Über Begriff und Gegenstand. *Vierteljahrsschrift für Wissenschaftliche Philosophie*, 16, 195–205. Translated as ‘On Concept and Object’ in Beaney (1997), pp. 181–93.
- Frege, G. (1892b). Über Sinn und Bedeutung. *Zeitschrift für Philosophie und philosophische Kritik*, 100, 25–50. “On Sense and Reference” translated in Frege (1952).
- Frege, G. (1952). In P. Geach, & M. Black (Eds.), *Translations from the philosophical writings of Gottlob Frege*. Basil Blackwell.
- Hodges, W. (2022). *Arabic logic and semantics*. Retrieved 2022, from wilfridhodges.co.uk
- Kamp, J. A. W. (1971). Formal properties of ‘now’. *Theoria*, 37.

- Kaplan, D. (1989). 'Demonstratives'. In J. Almog, J. Perry, & H. Wettstein (Eds.), *Themes from Kaplan* (pp. 481–564). Oxford University Press.
- Knuuttila, S. (1993). *Modalities in medieval philosophy*. Routledge.
- Lagerlund, H. (2016). Medieval theories of the syllogism. In E. N. Zalta (Ed.), *Stanford encyclopedia of philosophy* (Spring 2016). Stockholm University.
- Novaes, C. D. (2003). A medieval reformulation of the *de dicto/de re* distinction. *Logica Yearbook*, 111–124.
- Prior, A. N. (1952). Modality *de dicto* and modality *de re*. *Theoria*, 18, 174–180.
- Prior, A. N. (1957). *Time and modality*. Clarendon Press.
- Prior, A. N. (1967). *Past, present and future*. Clarendon Press.
- Prior, A. N. (1968). *Papers on time and tense*. Clarendon Press.
- Prior, A. N. (1971). In P. T. Geach, & A. J. P. Kenny (Eds.), *Objects of thought*. Oxford University Press.
- Quine, W. V. O. (1943). Notes on existence and necessity. *The Journal of Philosophy*, 40, 113–127.
- Quine, W. V. O. (1950). *Methods of logic*. Holt.
- Russell, B. (1905). On denoting. *Mind*, 14, 479–493. Reprinted in Robert C. Marsh, ed., *Logic and Knowledge: Essays 1901–1950, by Bertrand Russell*, Allen & Unwin, London, 1956.
- Smullyan, A. F. (1948). Modality and description. *The Journal of Symbolic Logic*, 13, 31–37.
- Stalnaker, R. & Thomason, R. (1968). Abstraction in first-order modal logic. *Theoria*, 34, 203–207.
- Street, T. (2004). Arabic logic. In D. M. Gabbay & J. Woods (Eds.), *Handbook of the history of logic* (Vol. 1, pp. 523–596). Elsevier North Holland.
- Thom, P. (2003). *Medieval modal systems, problems and concepts*. Ashgate.
- Thomason, R. & Stalnaker, R. (1968). Modality and reference. *Nous*, 2, 359–372.
- Vlach, F. (1973). 'Now' and 'Then': A formal study in the logic of tense anaphora. Doctoral dissertation, University of California, Los Angeles.
- von Wright, G. H. (1951). In L. E. J. Brouwer, E. W. Beth, & A. Heyting (Eds.), *An essay in modal logic*. Studies in Logic and the Foundations of Mathematics. North-Holland.
- Whitehead, A. N. & Russell, B. (1925–1927). *Principia mathematica* (Three volumes, 2nd ed.). Cambridge University Press.

Chapter 14

Predicate Abstraction, Formally



Predicate abstracts help sort out complications brought by non-rigid terms. In this chapter we present the formal machinery for predicate abstracts, and explore the application of these abstracts to non-rigid constant symbols, and also to non-rigid function symbols. And for both constant and function symbols, they might always designate, or there may be situations in which they do not. The main division is not between constants and functions—after all, a constant can be thought of as a function that takes no inputs. Rather, the division is between always designating and sometimes not doing so. We treat the always case first since it is simpler. Then we modify it to situations where non-designation could happen. We discuss constant symbols before bringing in the more complex function symbols.

The work here is of interest for its own sake, but it also serves as a kind of lead-in to our presentation of definite descriptions in Chap. 20. Definite descriptions are also non-rigid, but unlike constant symbols and function symbols, they have a kind of ‘meaning’ built into them. This meaning is used to pick out the object a definite description designates at each possible world where it designates. Generally, when issues of meaning and designation are involved, the term *intensional* is used, and we will sometimes do so here. We will also extend the use of *intensional* to refer not just to definite descriptions, once we get to them, but also to non-rigid constant and function symbols. Collectively we sometimes refer to them all as *intensional terms*.

Recall that our basic first-order modal logics came in two versions in this book: constant domains and varying domains. For each of these we have intensional terms that must always designate, and intensional terms that need not. Altogether we have four different versions of quantified modal logic admitting predicate abstracts. Of course we also have a variety of basic modal logics, **K**, **T**, **S4**, and so on, but the quantificational issues are fairly uniform across the range. Consequently most of our examples will involve **K**, since it is the simplest and is contained in all the others.

In the first edition of this book some of the details of predicate abstraction were noticeably different than they are here. The points being made are the same, but we feel the presentation is now smoother and some sticky points have been eliminated.

This is being said for those already familiar with the first edition, and newcomers can safely ignore the issue

The idea of introducing predicate abstraction into modal logic is from Thomason and Stalnaker (1968) and Stalnaker and Thomason (1968). Bressan (1972) gave an extensive development involving a treatment of higher-order modal logic, but the work is complex and has not had the influence it deserves. Details of the specific version presented here evolved through a series of papers, which we list for the record: Fitting (1972, 1973, 1975, 1991, 1993, 1996, 1998, 2002a, 2003, 2004, 2017, 2006, 2007). We also note that Fitting (2002b) extends the formalism to higher types and uses it to address Gödel’s ontological argument.

14.1 Constant Symbol Syntax

From here on we assume we have available in our formal modal language a countably infinite list of *constant symbols*, disjoint from any of the symbol alphabets introduced earlier. Informally we will use c , d and the like for this purpose. Their behavior at each possible world will be fixed, but it can vary from world to world, so we sometimes refer to these as *intensional constant symbols*.

We keep the definition of atomic formula exactly as it was in Chap. 8. An atomic formula is still an expression of the form $R(x_1, \dots, x_n)$, where R is an n -place relation symbol and $x_1, \dots, x_n$ are variables. In classical logic constant symbols are allowed to appear in atomic formulas, but we have only allowed variables here since doing otherwise modally can lead to ambiguous readings, as we have seen.

We now make an addition to our earlier definition of modal formula, and the numbering continues that of the earlier definition.

Definition 14.1.1 (Formulas With Constant Symbols) The following clauses are added to Definition 8.1.2:

6. Let Φ be a formula, x be a variable, and c be an intensional constant symbol.
 - a. $\langle \lambda x. \Phi \rangle$ is a *predicate abstract*; the free variable occurrences of $\langle \lambda x. \Phi \rangle$ are those of Φ except for occurrences of x .
 - b. $\langle \lambda x. \Phi \rangle(c)$ is a formula; the free variable occurrences of $\langle \lambda x. \Phi \rangle(c)$ are those of the predicate abstract $\langle \lambda x. \Phi \rangle$.

For example, the following is a formula and exactly one variable occurrence in it is free: $(\forall y) \langle \lambda x. P(x, y, z) \rangle(c)$. It is intended that $\langle \lambda x. \Phi \rangle$ be thought of as the predicate “abstracted” from the formula Φ , and then $\langle \lambda x. \Phi \rangle(c)$ should be read, *the object designated by c has the property specified by the predicate $\langle \lambda x. \Phi \rangle$* .

As has been the case in earlier chapters, some of our arguments will involve Complete Induction on formula complexity, measured by degree. Our earlier version, Definition 8.1.4, is updated to the following.

Definition 14.1.2 (Degree) The degree of a first-order modal formula allowing predicate abstraction is the total number of occurrences of $\neg$, $\wedge$, $\vee$, $\supset$, $\square$, $\diamond$, $\forall$, $\exists$ and λ in it.

As we will see, formulas involving predicate abstraction can get notationally complex and consequently difficult to read. It is occasionally convenient to make use of the following device.

Definition 14.1.3 (Abbreviated Iterated Predicate Abstracts) Directly nested predicate abstracts can be abbreviated, writing $\langle \lambda x_1, x_2, x_3. \Phi \rangle (c_1, c_2, c_3)$ in place of $\langle \lambda x_1. \langle \lambda x_2. \langle \lambda x_3. \Phi \rangle (c_3) \rangle (c_2) \rangle (c_1)$, for instance. For the record, a proper definition of this uses the following recursive scheme.

$$\langle \lambda y_1, y_2, \dots. \Phi \rangle (d_1, d_2, \dots) = \langle \lambda y_1. \langle \lambda y_2, \dots. \Phi \rangle (d_2, \dots) \rangle (d_1)$$

The following illustrates how the recursive scheme works.

$$\begin{aligned} \langle \lambda x_1, x_2, x_3. \Phi \rangle (c_1, c_2, c_3) &= \langle \lambda x_1. \langle \lambda x_2, x_3. \Phi \rangle (c_2, c_3) \rangle (c_1) \\ &= \langle \lambda x_1. \langle \lambda x_2. \langle \lambda x_3. \Phi \rangle (c_3) \rangle (c_2) \rangle (c_1) \end{aligned}$$

14.2 Constant Symbol Semantics, Always Designate Case

We have added intensional constant symbols to the language, so we must give meaning to them in models. Constant symbols should designate objects, but since we will allow them to be non-rigid, reference is allowed to change from world to world. We are now assuming that constant symbols always designate. The possibility of their not doing so will be addressed starting with Sect. 14.5.

Definition 14.2.1 (Interpretation, Always Designate) $\mathcal{I}$ is an interpretation in a skeleton $\mathcal{F} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$, allowing non-rigid intensional constant symbols that always designate, if:

1. $\mathcal{I}$ meets the conditions of Definitions 8.6.2 (constant domains) or 8.7.3 (varying domains) assigning relations to relation symbols, and also
2. $\mathcal{I}$ meets the condition that to each intensional constant symbol c , and to each $\Gamma \in \mathcal{G}$, $\mathcal{I}$ assigns some member of the domain of the frame. That is, $\mathcal{I}(c, \Gamma) \in \mathcal{D}(\mathcal{F})$.

One can read $\mathcal{I}(c, \Gamma)$ as “ c at Γ ”. It can, of course, happen that the value of an intensional constant symbol at a world of a varying domain model might not be something actually in the domain of that world. We are explicitly allowing constant symbols to designate nonexistent objects, though the objects designated must exist at *some* world. (Notice that this becomes moot if we are dealing with a constant domain model, for then existence anywhere is existence everywhere.)

As an example, consider the definite description “the first President of the United States.” (We will treat definite descriptions formally in Chap. 20. For now, intuition will suffice.) Using the natural temporal reading, this definite description designates George Washington: it designates him now, in this world, even though he does not now exist.

The following extends Definitions 8.6.3 and 8.7.4 to allow non-rigid constant symbols and predicate abstracts.

Definition 14.2.2 (Model, Always Designate) A model, allowing constant symbols that always designate, is a structure $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ where $\langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ is a skeleton and $\mathcal{I}$ is an interpretation in it that allows non-rigid intensional constant symbols.

Finally, the main event: extending the definition of truth to take intensional constants into account. We note that this will be modified significantly in Definition 14.6.2.

Definition 14.2.3 (Truth, Always Designate) Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a model allowing intensional constant symbols that always designate. The definition of $\mathcal{M}, \Gamma \Vdash_v \Phi$ is exactly as in Definitions 8.6.7 and 8.7.8, with the addition of one more clause:

10. For an intensional constant symbol c that always designates,

$$\mathcal{M}, \Gamma \Vdash_v \langle \lambda x. \Phi \rangle (c) \iff \mathcal{M}, \Gamma \Vdash_w \Phi,$$

where w is the x -variant of v such that $w(x) = \mathcal{I}(c, \Gamma)$.

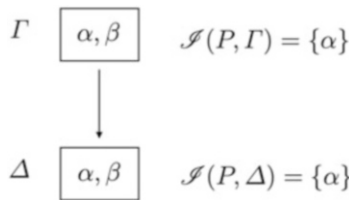
That is, $\langle \lambda x. \Phi \rangle (c)$ is true at Γ if Φ turns out to be true at Γ when we assign to x whatever it is that c designates at the world Γ .

Example 14.2.4 We return to Example 13.1.1, applying the machinery we have now developed. In that example we specified an interpretation of the constant symbol c informally. Now it becomes a formal part of the definition of our model, by the addition of the following conditions on the interpretation $\mathcal{I}$.

$$\mathcal{I}(c, \Gamma) = \alpha$$

$$\mathcal{I}(c, \Delta) = \beta$$

We repeat our earlier picture of the situation.



According to Definition 14.2.3, $\mathcal{M}, \Gamma \Vdash_v \langle \lambda x. \Diamond P(x) \rangle(c)$ is equivalent to $\mathcal{M}, \Gamma \Vdash_w \Diamond P(x)$, where w is the x -variant of v such that $w(x) = \mathcal{I}(c, \Gamma) = \alpha$. Since Δ is the only world in the model accessible from Γ , this in turn is equivalent to $\mathcal{M}, \Delta \Vdash_w P(x)$, which is equivalent to $w(x) \in \mathcal{I}(P, \Delta)$. Since indeed, $\alpha \in \mathcal{I}(P, \Delta)$, $\langle \lambda x. \Diamond P(x) \rangle(c)$ is *true* at Γ .

Next, $\mathcal{M}, \Gamma \Vdash_v \Diamond \langle \lambda x. P(x) \rangle(c)$ is equivalent to $\mathcal{M}, \Delta \Vdash_v \langle \lambda x. P(x) \rangle(c)$. By condition 10 of Definition 14.2.3, this is equivalent to $\mathcal{M}, \Delta \Vdash_w P(x)$ where $w(x) = \mathcal{I}(c, \Delta) = \beta$. This is further equivalent to $w(x) \in \mathcal{I}(P, \Delta)$, which is not the case since $\beta \notin \mathcal{I}(P, \Delta)$. Thus $\Diamond \langle \lambda x. P(x) \rangle(c)$ is *false* at Γ .

Predicate abstraction allows us to separate the ambiguous and unofficial formula $\Diamond P(c)$ into two distinct formulas, $\langle \lambda x. \Diamond P(x) \rangle(c)$ and $\Diamond \langle \lambda x. P(x) \rangle(c)$, and these are, indeed, not equivalent. The example above shows that $\langle \lambda x. \Diamond P(x) \rangle(c) \supset \Diamond \langle \lambda x. P(x) \rangle(c)$ is not valid. Exercise 14.2.1 asks you to show the converse is not valid either.

One last remark before we move on. We have just added predicate abstraction to a *quantified* modal logic. The presence of quantifiers already gives us logics with no decision procedure, by Church's Theorem. But we could have added to *propositional* modal logics the machinery of predicates and free variables, *but without quantifiers*, and also add predicate abstraction. In some ways such a logic can be thought of as between propositional and quantified. The situation now is more complicated. For **K**, **T**, and **D**, the result is a decidable logic. For **S5** it is undecidable. And for logics between **K4** and **S5** we have undecidability provided equality is present. This was shown in Fitting (2002a).

Exercises

Exercise 14.2.1 For this question there are no modal conditions so the underlying logic is **K**. It is assumed that constant symbols always designate.

1. Show there is a constant domain model where $\Diamond \langle \lambda x. P(x) \rangle(c) \supset \langle \lambda x. \Diamond P(x) \rangle(c)$ is not valid.
2. Show that $(\forall x) \Diamond P(x) \supset \langle \lambda x. \Diamond P(x) \rangle(c)$ is valid in all constant domain models.
3. Show that $(\forall x) \Diamond P(x) \supset \langle \lambda x. \Diamond P(x) \rangle(c)$ is not valid in some varying domain model.
4. Show that $(\forall x) \Diamond P(x) \supset \Diamond \langle \lambda x. P(x) \rangle(c)$ is not valid in some constant domain model.

Exercise 14.2.2 Show the validity, in all varying domain **K** models, of the following, where constant symbols always designate.

1. $\langle \lambda x. \neg \Phi \rangle(c) \equiv \neg \langle \lambda x. \Phi \rangle(c)$.
2. $\langle \lambda x. (\Phi \supset \Psi) \rangle(c) \equiv ((\langle \lambda x. \Phi \rangle(c) \supset \langle \lambda x. \Psi \rangle(c)))$.
3. $\langle \lambda x. (\Phi \wedge \Psi) \rangle(c) \equiv ((\langle \lambda x. \Phi \rangle(c) \wedge \langle \lambda x. \Psi \rangle(c)))$.
4. $\langle \lambda x. (\Phi \vee \Psi) \rangle(c) \equiv ((\langle \lambda x. \Phi \rangle(c) \vee \langle \lambda x. \Psi \rangle(c)))$.

14.3 Function Symbol Syntax

People often remember functions from mathematics: the *sine* function, the *square* function, and so on. But we commonly make use of functions in everyday life, though usually without the machinery of special symbols to denote them: the *age* function mapping each person to their age, the *father-of* function, mapping each person to their father, and so on. But like constants, everyday functions can be non-rigid in the sense that their behavior can be different at different possible worlds. For instance, a tide table can be used to map a date and a time of day to the time of the next high tide after that date and time. But different places on the edge of the ocean require different tide tables. You need to say what city on the coast you are interested in, and that determines what tide table to use. We can then think of *next-high-tide-after* as a non-rigid function whose behavior varies with a choice of location, and we can model a location as a kind of possible world. We will return to this in Example 14.4.5.

As another geographical example, stores in the United States often charge *sales tax* on purchases. Cash registers come with programs to automatically compute sales tax. But different cities may have different tax rates. We can think of cities as possible worlds, and talk about the non-rigid *sales-tax* function, whose exact behavior depends on what city we are in.

We will see more examples as we go on. Of course many functions, such as those of mathematics, are rigid. But we will do as we did with constants, and think of rigid functions as special cases of non-rigid functions—functions whose behavior does not change with a change in possible world.

Just as with relation symbols, we assume we have available a countably infinite list of *one place function symbols*, a disjoint countably infinite list of *two place function symbols*, a countably infinite list of *three place function symbols* disjoint from the first two lists, and so on. In practice we will be informal in our notation, and use f , g , or something similar for a function symbol, with its arity determined from context. We will think of these as representing *non-rigid functions*. Constant symbols are sometimes taken to be function symbols of arity 0, and you can do this if you want.

In mathematics, nesting function symbols is standard. For instance, if f is a 3-place function symbol, g is a 2-place function symbol, x and y are variables, and c and d are constant symbols, then $f(x, g(c, x), g(y, g(c, d)))$ would be a meaningful expression. We will not allow this here because of the same kind of scope issues that arose for intensional constant symbols. For instance, if P is a one place predicate symbol, f is a one place function symbol, and c is a constant symbol, how should we read the following?

$$\Diamond P(f(c)) \tag{14.1}$$

What is the scope associated with f , what with c , are they the same, are they different? How do these scopes relate to where P is evaluated? The expression is

ambiguous. We do not allow it, but we will see various correct counterparts of it in (14.2). The approach formally adopted here makes scopes explicit, at the cost of complicating readability, but with the advantage of avoiding ambiguity.

Definition 14.3.1 (Formulas With Constant and Function Symbols)

6. Definition 14.1.1 is replaced with the following extension of Definition 8.1.2.

- a. $\langle \lambda x. \Phi \rangle$ is a *predicate abstract*; the free variable occurrences of $\langle \lambda x. \Phi \rangle$ are those of Φ except for occurrences of x .
- b. A constant symbol c is an *intensional term*. If f is an n -ary function symbol and $x_1, \dots, x_n$ are variables, $f(x_1, \dots, x_n)$ is an *intensional term*.
- c. If t is an intensional term, then $\langle \lambda x. \Phi \rangle(t)$ is a formula; the free variable occurrences of this formula are those of the predicate abstract $\langle \lambda x. \Phi \rangle$, together with all variable occurrences in t .

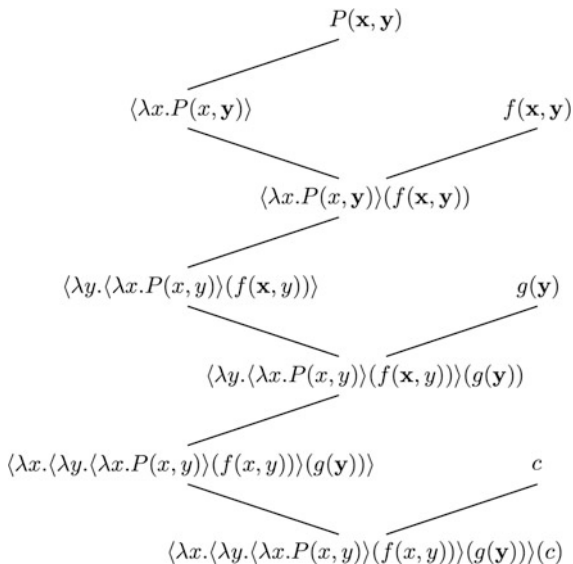
Now let's return to the earlier example of (14.1), $\Diamond P(f(c))$, which we said we would not allow. A proper version of it using predicate abstraction could be any of the following.

$$\begin{aligned}
 &\langle \lambda x. \langle \lambda y. \Diamond P(y) \rangle (f(x)) \rangle (c) \\
 &\langle \lambda x. \Diamond \langle \lambda y. P(y) \rangle (f(x)) \rangle (c) \\
 &\Diamond \langle \lambda x. \langle \lambda y. P(y) \rangle (f(x)) \rangle (c)
 \end{aligned} \tag{14.2}$$

An unfortunate aspect of this notation is that it is hard to read, and it might not be clear that these are all versions of $\Diamond P(f(c))$. But this can be seen informally as follows. Take one of these formulas, say the first, ignore the varying position possibility operator getting $\langle \lambda x. \langle \lambda y. P(y) \rangle (f(x)) \rangle (c)$, and now ‘unravel’ it. The λx at the beginning goes with the c at the end. Get rid of both, and then substitute occurrences of c for free occurrences of x in what remains. This process is, in a sense, what λ abstraction is telling us to do. This gets us $\langle \lambda y. P(y) \rangle (f(c))$. Now repeat this, replacing free occurrences of y by occurrences of $f(c)$. This gets us $P(f(c))$. Finally put the possibility operator back, getting $\Diamond P(f(c))$. Trivially, doing this to any of the three formulas of (14.2) will get us this same expression. Eliminating the abstraction machinery in this way gives us what looks like a more palatable version, but along the way information has been lost. We will see that semantically, the formulas of (14.2) do not all have the same meaning. What we have given up in reading ease has gained us the ability to formally express nuances that otherwise must be lost. We have gained clarity of meaning at the cost of complexity of notation.

Example 14.3.2 This is an intentionally misleading example. Our need for predicate abstracts involving function symbols will not actually require much complexity because things encountered in real life tend to be ‘shallow’ so to speak. Nonetheless, the definition is a recursive one and so the complexity could be arbitrarily great. We show how the definition above accommodates this complexity by creating an

Fig. 14.1 Building with predicate abstracts



expression example consisting of nested predicate abstracts. It will probably come as a relief to know that this is a more complicated item than anything you're ever likely to run into. We are just illustrating how the machinery fits together.

Assume P is a two place relation symbol, c is a constant symbol, f is a two place function symbol, and g is a one place function symbol. We use our machinery to show the following is a proper construct.

$$\langle \lambda x. \langle \lambda y. \langle \lambda x. P(x, y) \rangle (f(x, y)) \rangle (g(y)) \rangle (c)$$

The construction is displayed graphically in Fig. 14.1. In it, the middle column is occupied by formulas that involve predicate abstracts. Each formula gives rise to a new predicate abstract, shown below and to its left. On the right are intensional terms. Each predicate abstract on the left, applied to the intensional term to its right gives rise to the formula below the two. Throughout, variable occurrences that are free are shown in bold face. The final predicate abstraction has a single free variable occurrence, of y . Note, for instance, that in the first appearance of $\langle \lambda x. P(x, y) \rangle (f(\mathbf{x}, y))$ the occurrence of x in $P(x, y)$ is bound while the occurrence $f(\mathbf{x}, y)$ is free.

14.4 Function Symbol Semantics, Always Designate Case

To see that the semantic behavior of the formulas of (14.2) are not all the same, we need a semantics. As it happens, this is almost in place already. Definition 14.2.1 must be supplemented to allow for the interpretation of non-rigid function symbols,

and some other modifications must be made to a few other earlier definitions, but it is all rather straightforward.

Definition 14.4.1 (Interpretation, Always Designate) Definition 14.2.1 is extended with the following additional requirement:

3. $\mathcal{I}$ meets the condition that to each n -ary function symbol f , and to each $\Gamma \in \mathcal{G}$, $\mathcal{I}$ assigns an n -ary function on the domain of the frame. More precisely, $\mathcal{I}(f, \Gamma) : [\mathcal{D}(\mathcal{F})]^n \rightarrow \mathcal{D}(\mathcal{F})$.

Definition 14.2.2 for models must also be extended, but the only change is that interpretations are now also required to be defined on intensional function symbols. We won't bother writing this out formally. Finally Definition 14.2.3 for truth in a model must be revised to take function symbols into account.

Definition 14.4.2 (Truth, Always Designate) Definition 14.2.3 is extended with one more clause.

11. For an n place function symbol f that always designates,

$$\mathcal{M}, \Gamma \Vdash_v \langle \lambda y. \Phi \rangle (f(x_1, \dots, x_n)) \iff \mathcal{M}, \Gamma \Vdash_w \Phi,$$

where w is the y -variant of v such that $w(y) = \mathcal{I}(f, \Gamma)(v(x_1), \dots, v(x_n))$.

So, at a possible world a predicate abstract applied to an intensional term involving a function symbol has us evaluate the function that the function symbol designates at that world, on the values of its inputs, and record the answer as the value of the variable the predicate abstract tells us to use.

In the rest of this section we examine some models that involve predicate abstracts using function symbols.

Example 14.4.3 We revisit the model described in Example 14.2.4, which in turn continues Example 13.1.1. We supplement that model by extending the interpretation to cover a one-place function symbol f . We then use the resulting model to evaluate the three formulas listed as (14.2). Here are the conditions we add to specify the behavior of f in the model.

$$\begin{array}{ll} \mathcal{I}(f, \Gamma)(\alpha) = \beta & \mathcal{I}(f, \Delta)(\alpha) = \alpha \\ \mathcal{I}(f, \Gamma)(\beta) = \alpha & \mathcal{I}(f, \Delta)(\beta) = \alpha \end{array}$$

That is, f is interpreted so that at Γ it represents the function that switches α and β around, and at Δ it represents the function mapping everything to α . We evaluate the three formulas of (14.2) at Γ in the model $\mathcal{M}$ described in the earlier Example, now supplemented with the interpretation of f just given. Actually, we do two of the calculations in detail, and leave one to you as an exercise. Each case begins with the use of an arbitrary valuation v .

1. The formula is $\langle \lambda x. \langle \lambda y. \Diamond P(y) \rangle (f(x)) \rangle (c)$.

$$\begin{aligned}
 \mathcal{M}, \Gamma \Vdash_v \langle \lambda x. \langle \lambda y. \Diamond P(y) \rangle (f(x)) \rangle (c) &\iff \mathcal{M}, \Gamma \Vdash_w \langle \lambda y. \Diamond P(y) \rangle (f(x)) \\
 &\text{where } w \text{ is like } v \text{ except that} \\
 w(x) = \mathcal{I}(c, \Gamma) &= \alpha \\
 &\iff \mathcal{M}, \Gamma \Vdash_u \Diamond P(y) \\
 &\text{where } u \text{ is like } w \text{ except that} \\
 u(y) = \mathcal{I}(f, \Gamma)(w(x)) &= \\
 \mathcal{I}(f, \Gamma)(\alpha) &= \beta \\
 &\iff \mathcal{M}, \Delta \Vdash_u P(y)
 \end{aligned}$$

and this is false because $u(y) = \beta$ and this is not in $\mathcal{I}(P, \Delta)$. (In Example 13.1.1 P was stipulated to hold of only α at both worlds Γ and Δ .)

2. The formula is $\langle \lambda x. \Diamond \langle \lambda y. P(y) \rangle (f(x)) \rangle (c)$. This is left as Exercise 14.4.1.
 3. The formula is $\Diamond \langle \lambda x. \langle \lambda y. P(y) \rangle (f(x)) \rangle (c)$.

$$\begin{aligned}
 \mathcal{M}, \Gamma \Vdash_v \Diamond \langle \lambda x. \langle \lambda y. P(y) \rangle (f(x)) \rangle (c) &\iff \mathcal{M}, \Delta \Vdash_v \langle \lambda x. \langle \lambda y. P(y) \rangle (f(x)) \rangle (c) \\
 &\iff \mathcal{M}, \Delta \Vdash_w \langle \lambda y. P(y) \rangle (f(x)) \\
 &\text{where } w \text{ is like } v \text{ except that} \\
 w(x) = \mathcal{I}(c, \Delta) &= \beta \\
 &\iff \mathcal{M}, \Delta \Vdash_u P(y) \\
 &\text{where } u \text{ is like } w \text{ except that} \\
 u(y) = \mathcal{I}(f, \Delta)(w(x)) &= \\
 \mathcal{I}(f, \Delta)(\beta) &= \alpha
 \end{aligned}$$

and this is true because $u(y) = \alpha$ and this is in $\mathcal{I}(P, \Delta)$.

Example 14.4.4 Here is an example that originated in Fitting (2017). Suppose we want to say, of a particular person named “Alice”, that she might have been taller than she is (say if she had eaten better breakfasts when growing up). We use a modal reading under which $\Diamond \Phi$ expresses that Φ is true in an alternative state to the way things actually are, perhaps a state having a different past. This is background intuition, and we needn’t dwell on the details. We can represent Alice by a constant symbol a . As a first attempt, we could introduce a two place relation symbol T with the idea that $T(x, y)$ is intended to mean that x is taller than y . Then a first attempt might be the following formula. It asserts that for what a designates as things are,

namely Alice, in a possible alternative world it could be the case that for Alice there, that Alice is taller than this Alice. But that simply won't work.

$$\langle \lambda y. \Diamond \langle \lambda x. T(x, y) \rangle (a) \rangle (a) \quad (14.3)$$

The problem is that “Alice” is a name and names are commonly understood to be *rigid* designators. In the modal model we are constructing, a should represent the same individual in the present world and in alternative circumstances. After all, it is Alice of whom we are saying that she might have been taller, so the value assigned to a should be the same across possible worlds. But then (14.3) becomes problematic. In effect we are saying Alice might be taller than herself.

In Chap. 19 we will look at rigidity more formally. There we will see that if a is rigid, the following is true.

$$\langle \lambda y. \Box \langle \lambda x. x = y \rangle (a) \rangle (a) \quad (14.4)$$

And from (14.3) and (14.4) we will see we have the following as a consequence.

$$\langle \lambda x. \Diamond T(x, x) \rangle (a) \quad (14.5)$$

and this is clearly wrong. It is the formalized version of saying that it is possible that Alice is taller than Alice.

There is a way around the problem however. When we say that Alice is represented by a rigid designator, a , what does that mean? In some sense a designates the same person in different possible worlds, but in different possible worlds that person might have different inessential properties. In particular, the height of a might be different under different circumstances, in different possible worlds, even though a is the same person.

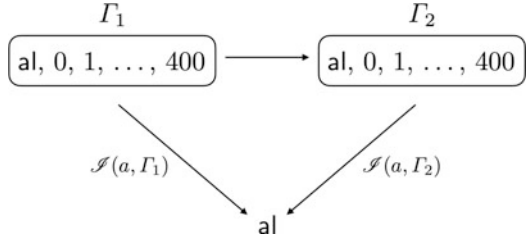
Suppose we introduce a *non-rigid function symbol*, h , where we can think of “ $h(a)$ ” as “the height of person a ”, where height is a number. What h assigns to a depends on the possible world in which $h(a)$ is evaluated. The point is that even though a is rigid, $h(a)$ can vary from possible world to possible world. This happens because an inessential attribute of a can vary from world to world, and this is reflected in the non-rigidity of h , rather than of a . If we assume G is the two-place greater-than relation on numbers, our assertion that Alice might have been taller than she is, formalizes as follows.

$$\langle \lambda y. \Diamond \langle \lambda x. G(x, y) \rangle (h(a)) \rangle (h(a)) \quad (14.6)$$

Of course this can't be quite right, since (14.6) is not a legal formula. We can't actually write $h(a)$. But we have handled this sort of thing before. We need the correct, but appalling looking formula (14.7), which has no function/constant symbol nesting.

$$\langle \lambda w. \langle \lambda y. \Diamond \langle \lambda z. \langle \lambda x. G(x, y) \rangle (h(z)) \rangle (a) \rangle (h(w)) \rangle (a) \quad (14.7)$$

Fig. 14.2 Model $\mathcal{N}$ with
alice



Actually, this can be simplified a bit. The constant symbol a is rigid. It has two appearances in (14.7) but, being rigid, a should always be the same object so we can manage with a single appearance. The following is, in fact, equivalent to (14.7), assuming rigidity. Note that the number of variables has been reduced; z is gone, and its role has been taken on by w .

$$\langle \lambda w. \langle \lambda y. \Diamond \langle \lambda x. G(x, y) \rangle (h(w)) \rangle (h(w)) \rangle (a) \quad (14.8)$$

We now construct a plausible model for formula (14.7), or equivalently (14.8). We don't actually need the whole real number system for our measurements, since we are only able to distinguish a finite set of actual measurements. So let's say, for simplicity, that we measure heights to the nearest centimeter, and that we only need heights from 0 to 400. Of course nothing essential depends on this particular choice. And we want the interpretation of a to be rigid, but the function interpreting h to be non-rigid.

Figure 14.2 shows a formal model, $\mathcal{N}$. The set of possible worlds $\mathcal{G}$ consists of Γ_1 and Γ_2 (how things are, and how they might have been), with $\Gamma_1 \mathcal{R} \Gamma_2$. The object domain $\mathcal{D}$ is $\{\text{al}, 0, 1, \dots, 400\}$, where **al** is intended to be the 'Alice object.'

For an interpretation, let G be interpreted as the greater-than relation,

$$\mathcal{J}(G, \Gamma_1) = \mathcal{J}(G, \Gamma_2) = \{(x, y) \mid x, y \text{ are integers from } 0 \text{ to } 400 \text{ and } x > y\}.$$

Intensional constant a designates at both Γ_1 and Γ_2 , and $\mathcal{J}(a, \Gamma_1) = \mathcal{J}(a, \Gamma_2) = \text{al}$. Thus a is interpreted rigidly. The height function h is interpreted non-rigidly, and designates at both possible worlds. Rather arbitrarily, we set $\mathcal{J}(h, \Gamma_1)(\text{al}) = 165$ and $\mathcal{J}(h, \Gamma_2)(\text{al}) = 180$. (Values on other members of $\mathcal{D}$ are not relevant, and we don't need to specify them.)

It can now be verified that we have

$$\mathcal{N}, \Gamma_1 \models_v \langle \lambda w. \langle \lambda y. \Diamond \langle \lambda x. G(x, y) \rangle (h(w)) \rangle (h(w)) \rangle (a)$$

and so (14.8) is satisfiable, pretty much as expected.

Example 14.4.5 At the beginning of Sect. 14.3 we mentioned *tide tables* as providing a case where non-rigid function symbols could be useful. We now consider an example formalizing exactly this scenario. Suppose we have a model

$\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ defined as follows. The set of possible worlds, $\mathcal{G}$, is the set of all locations on Earth for which tide tables are published. The accessibility relation, $\mathcal{R}$, is universal, every possible world is accessible from every possible world. The domain function is constant, that is, we have a constant domain model, and it consists of all times starting at some arbitrary value (say midnight, January 1, 2000, in London) and continuing into the future. We can assume these are represented using UTC (coordinated universal time), so that the representation is independent of place. We ignore issues of the solar system ending, the ocean drying up, and the like. (This is a make-believe example.) We assume our formal language has just one relation symbol, which we write in infix position, $x < y$. Our interpretation of it is independent of possible world. It says the UTC value assigned to x is earlier than the UTC value assigned to y .

So far we have discussed general background. Now we bring tide tables and non-rigidity into things. Assume we have a non-rigid unary function symbol f , interpreted to be a mapping from UTC codes to UTC codes, such that at any possible world Γ : $f(x)$ is the UTC code for the time of the next high tide following time x , where we determine this using the tide table for the possible world (coastal location) Γ .

We want a formula Φ with no free variables, but containing a rigid constant symbol c , such that when evaluated at a possible world Γ with c interpreted to be the current time, Φ ‘says’ that somewhere the next high tide will occur earlier than it does at Γ . And we claim the formula $\Phi = \langle \lambda x. \langle \lambda y. \Diamond \langle \lambda z. z < y \rangle (f(x)) \rangle (f(x)) \rangle (c)$ will do this. Here is a sketch of the verification. In the following, Γ is an arbitrary possible world of model $\mathcal{M}$ and v_1 is an arbitrary valuation. The rigid constant symbol is interpreted to be the UTC code of the current moment; we’ll simply refer to this as *now*.

1. $\mathcal{M}, \Gamma \Vdash_{v_1} \langle \lambda x. \langle \lambda y. \Diamond \langle \lambda z. z < y \rangle (f(x)) \rangle (f(x)) \rangle (c)$ is equivalent to $\mathcal{M}, \Gamma \Vdash_{v_2} \langle \lambda y. \Diamond \langle \lambda z. z < y \rangle (f(x)) \rangle (f(x))$, where v_2 is the x variant of v_1 assigning *now* to x .
2. $\mathcal{M}, \Gamma \Vdash_{v_2} \langle \lambda y. \Diamond \langle \lambda z. z < y \rangle (f(x)) \rangle (f(x))$ is equivalent to $\mathcal{M}, \Gamma \Vdash_{v_3} \Diamond \langle \lambda z. z < y \rangle (f(x))$, where v_3 is the y -variant of v_2 assigning $\mathcal{I}(f, \Gamma)(v_2(x))$ to y . Since $v_2(x)$ is *now*, and at Γ the interpretation $\mathcal{I}$ uses the tide table for Γ , $v_3(y)$ is the time of the next high tide at Γ .
3. $\mathcal{M}, \Gamma \Vdash_{v_3} \Diamond \langle \lambda z. z < y \rangle (f(x))$ is equivalent to $\mathcal{M}, \Delta \Vdash_{v_3} \langle \lambda z. z < y \rangle (f(x))$ for some Δ .
4. $\mathcal{M}, \Delta \Vdash_{v_3} \langle \lambda z. z < y \rangle (f(x))$ for some Δ is equivalent to $\mathcal{M}, \Delta \Vdash_{v_4} z < y$ for some Δ where, just as in the previous step, v_4 is the z -variant of v_3 , where $v_4(z)$ is the time of the next high tide at Δ .
5. $\mathcal{M}, \Delta \Vdash_{v_4} z < y$ for some Δ , using the interpretation $\mathcal{I}$ and what v_4 assigns to z and y , says that for some Δ , the time of the next high tide at Δ is earlier than the time of the next high tide at Γ .

Exercises

Exercise 14.4.1 Evaluate the middle case for Example 14.4.3.

Exercise 14.4.2 Use the semantic machinery of Example 14.4.5 except that the accessibility relation is changed so that every possible world is accessible from every *other* possible world, but not from itself. Write a formula, similar to the one we wrote, that is true in a possible world Γ just in case no other possible world has the same time for its next high tide that Γ does.

14.5 Partiality and Designation

So far, under an interpretation in a model constant symbols designate some object at each possible world. Actually, this is an unnatural requirement—it is really quite strong. For example, suppose we want to create a modal model in which the possible worlds are dates throughout the history of our planet Earth, and we want to consider a constant symbol t intended to denote the tallest person at any given date. What do we do with the vast period when there were no people? There is nobody for the symbol t to designate. As a specific question to give some thought to, should t always be self-identical even when there are no people? If not, what should be the status of $t = t$ under such circumstances? We could just say that when there are no people, by stipulation a formula involving t is simply false. But this would make both $t = t$ and $\neg(t = t)$ false, and thus destroy the classicality of negation. Or we could take such formulas to have a third truth value, call it *undefined*. But this too moves us out of the classical logic we have been using. There are many similar problems, involving all the connectives and quantifiers.

Actually, $t = t$ is not a proper formula in our formal system. Constant symbols cannot appear in atomic formulas as we have set things up. We should be writing $\langle \lambda x. x = x \rangle(t)$. But then the scoping mechanism of predicate abstraction gives us something we can work with. We can simply stipulate that a predicate abstract, when applied to a non-designating intensional term, is false. That is, if t doesn't designate, it is false to say that any particular property, as specified by a predicate abstract, applies to what it designates. This would make both $\langle \lambda x. x = x \rangle(t)$ and $\langle \lambda x. \neg(x = x) \rangle(t)$ false, of course. But $\langle \lambda x. \neg(x = x) \rangle(t)$ is not the same formula as $\neg \langle \lambda x. x = x \rangle(t)$ which should simply evaluate to true, since $\langle \lambda x. x = x \rangle(t)$ is false. We would wind up making a distinction between a negatively specified predicate abstract, and the negation of a positively specified predicate abstract. Specifically, we distinguish between having the non-self-identical property and not having the self-identical property. A non-designating term doesn't designate something that has the non-self-identical property, but it also doesn't designate something that has the self-identical property, so it is true that it is not the case that it does. Note that this is different behavior from what we saw in Exercise 14.2.2, but that was subject to the assumption that constant symbols always designated.

There is a natural confusion between existence and designation, but these are really orthogonal issues. Terms designate; objects exist. For instance the phrase “the first President of the United States” designates George Washington, though thinking temporally, the person being designated is no longer with us—the person designated does not exist, though he once did. The nonexistent George Washington is designated *now* by the phrase. On the other hand the phrase, “the present King of France,” does not designate anybody now, living or dead, though at certain past instances it did designate.

In our formal treatment we have allowed terms to designate nonexistent objects at worlds of varying domain models. More properly, a term may designate, at a world, an object not in the domain of that world, though it must be in the domain of *some* world. But up till now we have adopted the convenient fiction that terms always do designate. The problem for this section, then, is to modify the formal machinery to allow for non-designating constant symbols, and similarly for function symbols.

As we said, we take the straightforward approach that the application of any predicate abstract to a non-designating constant symbol yields a false sentence. A constant symbol that designates can designate an object that does not exist, or it can designate one that does. But since we can ascribe no properties to what is designated by a constant symbol that does not designate, we will see that we cannot correctly assert either the existence or the nonexistence of an object designated by a term that does not designate. We are thinking here of nonexistence as a positive property in its own right, rather than as the lack of the existence property. Careful attention to such matters will allow us to avoid the usual paradoxes. Now the details.

14.6 Non-Designation Formally

The notion of an interpretation allowing non-rigidity was given in Definition 14.2.1 for constant symbols, and extended to function symbols in Definition 14.4.1, under the assumption that constant and function symbols always designated. We now modify this to allow for *partiality*.

Definition 14.6.1 (Interpretation, Allowing Partiality) $\mathcal{I}$ is an interpretation in a skeleton $\mathcal{F} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ allowing partially defined intensional constant and function symbols, if:

1. $\mathcal{I}$ meets the conditions of Definitions 8.6.2 (constant domains) or 8.7.3 (varying domains) assigning relations to relation symbols,
2. $\mathcal{I}$ meets the condition that to each intensional constant symbol c , and to *some* (possibly no) members $\Gamma \in \mathcal{G}$, $\mathcal{I}$ assigns a member of the domain of $\mathcal{F}$. That is, $\mathcal{I}(c, \Gamma)$ may not be defined for particular possible worlds Γ , but if it is then $\mathcal{I}(c, \Gamma) \in \mathcal{D}(\mathcal{F})$, the domain of the skeleton. If $\mathcal{I}(c, \Gamma)$ is defined, we say that c *designates* at Γ .
3. $\mathcal{I}$ meets the condition that to each n -ary function symbol f , and to *some* $\Gamma \in \mathcal{G}$, $\mathcal{I}$ assigns an n -ary *partial* function on the domain of the frame. That is, $\mathcal{I}(f, \Gamma)$

may not be defined for some possible worlds Γ , but if it is, it is a function from a subset, not necessarily all, of $\mathcal{D}(\mathcal{F})^n$ to $\mathcal{D}(\mathcal{F})$. If $\mathcal{I}(f, \Gamma)$ is defined, we say that f *designates* at Γ . If f designates at Γ and $\langle d_1, \dots, d_n \rangle$ is in the domain of $\mathcal{I}(f, \Gamma)$, we say f is *specified* or *defined* on $\langle d_1, \dots, d_n \rangle$ at Γ .

Note that function symbols may not specify functions, under an interpretation, at some possible worlds, but if they do specify, they specify *partial* functions. That is, they may not be defined on some inputs. This is actually a familiar thing. The *sine* function, for example, is specified on all real numbers. It is not specified on cats and dogs. Likewise the *father-of* function is specified on cats and dogs, but not on real numbers.

The definition of model, allowing partiality, is the obvious one. We simply make use of the notion of interpretation as just defined. The key item is to modify our satisfaction and truth definitions to allow for non-designation.

Definition 14.6.2 (Truth in a Model, Allowing Partiality) Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a model, where the interpretation $\mathcal{I}$ allows partiality. The definition of $\mathcal{M}, \Gamma \Vdash_v \Phi$ remains the same as in Definitions 14.2.3 and 14.4.2, but with the following replacement for the predicate abstract clause.

10. For an intensional constant symbol c , if c designates at Γ in $\mathcal{G}$ then

$$\mathcal{M}, \Gamma \Vdash_v \langle \lambda x. \Phi \rangle(c) \iff \mathcal{M}, \Gamma \Vdash_w \Phi,$$

where w is the x -variant of v such that $w(x) = \mathcal{I}(c, \Gamma)$, and otherwise,

$$\mathcal{M}, \Gamma \nVdash_v \langle \lambda x. \Phi \rangle(c).$$

11. For an intensional n -place function symbol f , if f designates at $\Gamma \in \mathcal{G}$ and f is specified on $\langle v(x_1), \dots, v(x_n) \rangle$ at Γ then

$$\mathcal{M}, \Gamma \Vdash_v \langle \lambda y. \Phi \rangle(f(x_1), \dots, x_n)) \iff \mathcal{M}, \Gamma \Vdash_w \Phi,$$

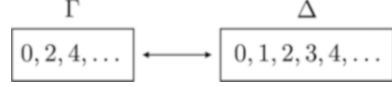
where w is the y -variant of v such that $w(y) = \mathcal{I}(f, \Gamma)(v(x_1), \dots, v(x_n))$, and otherwise

$$\mathcal{M}, \Gamma \nVdash_v \langle \lambda y. \Phi \rangle(f(x_1), \dots, x_n)).$$

Non-rigid function symbols are less common in the modal literature than are non-rigid constant symbols. Essentially this is because the fundamental philosophical points can be made using constant symbols, so why complicate things. Nonetheless, the machinery is intuitive and can be useful. We conclude this section with two examples, one quite artificial, the other still artificial, but not as much.

Example 14.6.3 In Fig. 14.3 we present the diagram of a modal model $\mathcal{M}$. It has no intrinsic interest, but does serve to illustrate the definitions just given. We work with

Fig. 14.3 Model illustrating non-rigidity



a formal language having two one-place predicate letters, E and P , two one-place function symbols, f and g , and a constant symbol, c . The accessibility relation, the worlds, and the domains of the worlds can be read off from the diagram. Here is the characterization of the interpretation function, where we have used informal but obvious notation to specify functions. Don't try to find any hidden meanings in what follows. This is just to illustrate how our machinery works.

$$\begin{aligned}
 \mathcal{I}(E, \Gamma) &= \{0, 2, 4, \dots\} \\
 \mathcal{I}(E, \Delta) &= \{0, 2, 4, \dots\} \\
 \mathcal{I}(P, \Gamma) &= \emptyset \\
 \mathcal{I}(P, \Delta) &= \{2, 3, 5, 7, 11, 13, \dots\} \\
 \mathcal{I}(f, \Gamma) &= n \rightarrow n + 1 \\
 \mathcal{I}(f, \Delta) &= n \rightarrow 2n \\
 \mathcal{I}(g, \Gamma) &= \text{undefined} \\
 \mathcal{I}(g, \Delta) &= n \rightarrow n + 1 \text{ if } n \neq 3, \text{ undefined at } 3 \\
 \mathcal{I}(c, \Gamma) &= 3 \\
 \mathcal{I}(c, \Delta) &= \text{undefined}
 \end{aligned}$$

Using all this, we now evaluate a few formulas to illustrate our machinery in use. More are given as exercises. This should be compared with Example 14.4.3, where terms always designated.

1. The formula $\langle \lambda x. \langle \lambda y. \Diamond P(y) \rangle (f(x)) \rangle (c)$ is false at Γ .

$$\begin{aligned}
 \mathcal{M}, \Gamma \Vdash_v \langle \lambda x. \langle \lambda y. \Diamond P(y) \rangle (f(x)) \rangle (c) &\iff \mathcal{M}, \Gamma \Vdash_w \langle \lambda y. \Diamond P(y) \rangle (f(x)) \\
 &\text{where } w \text{ is like } v \text{ except that} \\
 &w(x) = \mathcal{I}(c, \Gamma) = 3 \\
 &\iff \mathcal{M}, \Gamma \Vdash_u \Diamond P(y) \\
 &\text{where } u \text{ is like } w \text{ except that} \\
 &u(y) = \mathcal{I}(f, \Gamma)(w(x)) \\
 &= 2w(x) = 6 \\
 &\iff \mathcal{M}, \Delta \Vdash_u P(y)
 \end{aligned}$$

and this is false because $u(y) = 6$ and $6 \notin \mathcal{I}(P, \Delta)$.

2. The formula $\langle \lambda x. \Diamond \langle \lambda y. P(y) \rangle (g(x)) \rangle (c)$ is false at Γ .

$$\begin{aligned} \mathcal{M}, \Gamma \Vdash_v \langle \lambda x. \Diamond \langle \lambda y. P(y) \rangle (g(x)) \rangle (c) &\iff \mathcal{M}, \Gamma \Vdash_w \Diamond \langle \lambda y. P(y) \rangle (g(x)) \\ &\text{where } w \text{ is like } v \text{ except that} \\ w(x) = \mathcal{I}(c, \Gamma) = 3 \\ &\iff \mathcal{M}, \Delta \Vdash_w \langle \lambda y. P(y) \rangle (g(x)) \end{aligned}$$

but g is partial at Δ and is not specified on $w(x) = 3$, so this is false.

3. The formula $\langle \lambda x. \langle \lambda y. \langle \lambda z. \Diamond P(z) \rangle (f(y)) \rangle (g(x)) \rangle (c)$ is false at Γ .

$$\begin{aligned} \mathcal{M}, \Gamma \Vdash_v \langle \lambda x. \langle \lambda y. \langle \lambda z. \Diamond P(z) \rangle (f(y)) \rangle (g(x)) \rangle (c) &\iff \\ \mathcal{M}, \Gamma \Vdash_w \langle \lambda y. \langle \lambda z. \Diamond P(z) \rangle (f(y)) \rangle (g(x)) & \\ \text{where } w \text{ is like } v \text{ except that} & \\ w(x) = \mathcal{I}(c, \Gamma) = 3 & \end{aligned}$$

but g does not designate at Γ , so this is false.

Exercise 14.6.2 contains additional formulas to be evaluated in this model.

Example 14.6.4 What follows is a rather detailed example, first appearing in Fitting (2017). The example makes use of the phrase “the Swedish monarch,” which we can think of as being non-rigid and designating different people at different times, and sometimes not designating anybody. (We did not use “the King of Sweden” since Sweden has had three ruling Queens, most notably Queen Christina, though the example would have worked even if we had, since then “the King of Sweden” simply would have designated at some times and been undefined at certain others because the monarch was entitled Queen.) Note that this example is fundamentally different from the “Alice” Example 14.4.4. That example involved a name, which is understood to be a rigid designator. The present example involves a title, which can apply to a range of people throughout history.

At any point over a long period of history it would make sense to say, “Someday the Swedish monarch might be taller than now.” It is the behavior of this assertion that we want to examine. Suppose we use a modal logic where we give the modal operator a temporal reading. Semantically, possible worlds in a model will be considered to be points in time. For two such points, $\Gamma \mathcal{R} \Delta$ represents that Δ is a possible future to Γ . We can take, as domains, sets of people. For this example it really does not matter whether we do this in a possibilist or an actualist way since we will not make use of quantifiers. We can represent the monarch using the non-rigid constant symbol m , designating different persons at different times, or nobody at some times too. Finally, suppose we have a two-place predicate symbol T with the intended reading: $T(x, y)$ is true if the values of both arguments are people, and the first is a taller person than the second. Without predicate abstraction, but using standard modal machinery, the best formalization we could manage for “Someday

the Swedish monarch might be taller than now” is the $\Diamond T(m, m)$, and this would be silly.

It isn’t the monarchy to which we ascribe height. It’s the person who is the monarch at a particular time. We want to compare what m designates now with what m designates at a future time. In $\Diamond T(m, m)$, the first m must be understood as designating a person now; the second as designating at a possible future. Predicate abstraction gives us the machinery to do that. We can express that someday the monarch will be taller than now as follows.

$$\langle \lambda y. \Diamond \langle \lambda x. T(x, y) \rangle (m) \rangle (m) \quad (14.9)$$

Interpreted at the present moment, this says it is true of the person who is actually monarch (the value of y) that at some possible future state the taller-than predicate will hold between the person who then holds the position (the value of x) and the present person. We use this formula to illustrate the way our semantics works. To make things very concrete, here are a few historical facts we can make use of in setting up three illustrative possible worlds.

Swedish royalty is generally understood to begin with Eric the Victorious, who died around 995. The first, vague, reference to a Swedish Monarch (to a King, actually) is in Tacitus, around the year 100. So we can plausibly assume that in the year 50 there simply was no Swedish monarch.

An examination of skeletons shows that, in the thirteenth century the average adult male Swede was 174.3 cm (68.6 in.). An accessible reference for this is Ward (2022). The Swedish King Magnus III Barnlock died in 1290, and we will assume his height was exactly average. We will represent him in what follows as **mb**.

Finally, King Carl XVI Gustaf, monarch at the time of the publication of Fitting (2017), was listed as 179 cm (70.5 in.) tall. We will use **cg** to represent him. He is taller than Magnus III, though the difference is small.

Figure 14.4 graphically shows a model, $\mathcal{M}$. For this model the set of possible worlds is $\mathcal{G} = \{\Gamma_{50}, \Gamma_{1289}, \Gamma_{2017}\}$, intended to represent states of the world in the years 50, 1289, and 2017. Accessibility is shown using right pointing arrows, and is intended to represent passage to a future state. You may assume transitivity, though it is not shown explicitly. The object domain for all worlds is $\mathcal{D} = \{\mathbf{mb}, \mathbf{cg}, \dots\}$, intended to represent the two Swedish monarchs discussed above, and whoever else

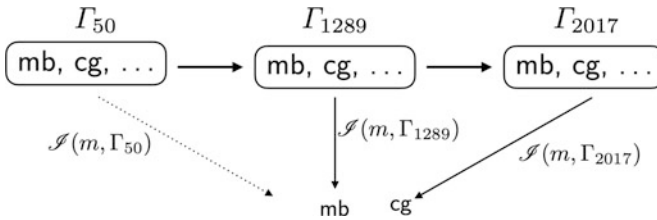


Fig. 14.4 Model $\mathcal{M}$ with Swedish monarchs

there might ever have been. The interpretation function $\mathcal{I}$ is such that $\mathcal{I}(T, \Gamma_{50}) = \mathcal{I}(T, \Gamma_{1289}) = \mathcal{I}(T, \Gamma_{2017}) = \{(\mathbf{cg}, \mathbf{mb})\}$. (Thus T is interpreted rigidly, and informally says that $\mathbf{cg}$ is taller than $\mathbf{mb}$. We don't care about the interpretation of T for anybody else that might exist, and so don't bother to mention it.) We assume m designates at Γ_{1289} and Γ_{2017} but not at Γ_{50} , and $\mathcal{I}(m, \Gamma_{1289}) = \mathbf{mb}$ and $\mathcal{I}(m, \Gamma_{2017}) = \mathbf{cg}$. That is, $\mathcal{I}$ picks out for m the King of Sweden at each time state we consider, provided a king exists. The people picked out are different at the two states where m designates.

We first check that $\mathcal{M}, \Gamma_{1289} \Vdash_v \langle \lambda y. \Diamond \langle \lambda x. T(x, y) \rangle (m) \rangle (m)$, thus showing that (14.9) is formally true under a circumstance where one would expect it to be. Here v is an arbitrary valuation; in fact, since the formula has no free variables, a choice of v doesn't matter. Note that the possibility operator jumps us from 1289 to 2017 because our model doesn't actually consider any states in between.

$$\begin{aligned}
 \mathcal{M}, \Gamma_{1289} \Vdash_v \langle \lambda y. \Diamond \langle \lambda x. T(x, y) \rangle (m) \rangle (m) &\iff \mathcal{M}, \Gamma_{1289} \Vdash_w \Diamond \langle \lambda x. T(x, y) \rangle (m) \\
 &\text{where } w \text{ is like } v \text{ except that} \\
 w(y) &= \mathcal{I}(m, \Gamma_{1289}) = \mathbf{mb} \\
 &\iff \mathcal{M}, \Gamma_{2017} \Vdash_w \langle \lambda x. T(x, y) \rangle (m) \\
 &\iff \mathcal{M}, \Gamma_{2017} \Vdash_u T(x, y) \\
 &\text{where } u \text{ is like } w \text{ except that} \\
 &\text{and } u(x) = \mathcal{I}(m, \Gamma_{2017}) = \mathbf{cg}
 \end{aligned}$$

Finally, $\mathcal{M}, \Gamma_{2017} \Vdash_u T(x, y)$ is true, because $(u(x), u(y)) = (\mathbf{cg}, \mathbf{mb}) \in \mathcal{I}(T, \Gamma_{2017})$.

Next $\mathcal{M}, \Gamma_{50} \nVdash_v \langle \lambda y. \Diamond \langle \lambda x. T(x, y) \rangle (m) \rangle (m)$, because m does not designate at Γ_{50} . This is what we would expect.

We would have $\mathcal{M}, \Gamma_{2017} \Vdash_v \langle \lambda y. \Diamond \langle \lambda x. T(x, y) \rangle (m) \rangle (m)$ if we had $\mathcal{M}, \Gamma_{2017} \Vdash_w \Diamond \langle \lambda x. T(x, y) \rangle (m)$ where w is like v except that $w(y) = \mathcal{I}(m, \Gamma_{2017})$. But we don't have this because Γ_{2017} is a possible world with no future in our model. That is, there are no accessible possible worlds, and so no possibility formulas can hold. This behavior really reflects an inadequacy in our model—we have not represented time correctly, since things did not end in 2017.

Exercises

Exercise 14.6.1 Let c be an intensional constant symbol.

1. Using the assumption that terms don't always designate, give a varying domain **K** model showing $(\forall x)P(x) \supset \langle \lambda x. P(x) \rangle (c)$ is not valid.

2. In Definition 15.4.3 we will introduce an existence predicate, which will then play an important role. Anticipating a bit, let $\mathbf{E}$ abbreviate the predicate abstract $\langle \lambda x. \mathbf{E}(x) \rangle$, where $\mathbf{E}(x)$ is from Definition 12.1.2 and abbreviates $(\exists y)(y = x)$. Show $[(\forall x)P(x) \wedge \mathbf{E}(c)] \supset \langle \lambda x. P(x) \rangle(c)$ is valid in all varying domain models.

Exercise 14.6.2 Evaluate the following using the model from Exercise 14.6.3.

1. $\mathcal{M}, \Gamma \Vdash_v \langle \lambda x. \Diamond \langle \lambda y. \Diamond \langle \lambda z. \Diamond P(z) \rangle (f(y)) \rangle (f(x)) \rangle (c)$
2. $\mathcal{M}, \Delta \Vdash_v \langle \lambda x. \Diamond \langle \lambda y. \Diamond \langle \lambda z. \Diamond P(z) \rangle (f(y)) \rangle (f(x)) \rangle (c)$
3. $\mathcal{M}, \Gamma \Vdash_v \langle \lambda x. \Diamond \langle \lambda y. \Diamond \langle \lambda z. P(z) \rangle (f(y)) \rangle (f(x)) \rangle (c)$
4. $\mathcal{M}, \Gamma \Vdash_v \langle \lambda x. \langle \lambda y. \Diamond \langle \lambda z. \Diamond P(z) \rangle (f(y)) \rangle (f(x)) \rangle (c)$

14.7 What We Still Can't Say

In the first edition of this book a somewhat different version of predicate abstraction with function symbols was used (constant symbols remain the same). For function symbols, things like (14.1) were allowed and the formal treatment was equivalent to identifying it with the third item of (14.2). Evaluation in a model, and tableau rules were more complicated. We could, of course, allow $\Diamond P(f(c))$ here and treat it as an abbreviation for $\Diamond \langle \lambda x. \langle \lambda y. P(y) \rangle (f(x)) \rangle (c)$, but it seems simpler to just require scope for each constant and function symbol to be expressed explicitly. This complicates the readability of formulas, but simplifies tableau rules. The result is more than sufficient for the examples we will discuss throughout the rest of the book. But it is not the end of the matter.

Consider the middle expression $\langle \lambda x. \Diamond \langle \lambda y. P(y) \rangle (f(x)) \rangle (c)$ from (14.2). To evaluate it at a possible world Γ of a model, we determine what object c designates at Γ , call it c_Γ , then the possibility operator moves us to an accessible world, say Δ , and we determine what function the function symbol f designates at Δ , call it f_Δ . We then apply f_Δ to c_Γ , getting the object $f_\Delta(c_\Gamma)$, and we see if it has the property corresponding to P in Δ , call it P_Δ . Note that we evaluate c at the initial world, but we evaluate f at an accessible world. If we want to reverse this order of evaluation, there is no way of expressing it.

To be a bit more concrete, we return to the tide tables of Example 14.4.5. Suppose c is a non-rigid constant symbol whose intended meaning is: at each coastal city (a possible world, in the example) c is the UTC value of noon at that city. And f is the city-dependent high tide function, so at a coastal city f tells us, given a UTC input, when is the next high tide after that input, for the particular coastal city. Then to evaluate $\langle \lambda x. \Diamond \langle \lambda y. P(y) \rangle (f(x)) \rangle (c)$ at a particular city, say Buenos Aires, c picks out the UTC value of noon at Buenos Aires—this is the value that gets assigned to x . Then the possibility operator moves us to some perhaps different coastal city, say to Canberra, and we use the tide table for Canberra for f . Then y is assigned the time of the next high tide after x , at Canberra. P , of course, is evaluated at Canberra. The formula evaluates to true if, at Canberra, the time of the next high tide after noontime at Buenos Aires has property P (as P is understood at Canberra). In this

scenario we are saying that the time of the high tide at Canberra that is next after noon at Buenos Aires has property P as determined at Canberra.

Now, suppose we wanted to reverse things, and apply the tide table from Buenos Aires to the UTC time of noon at Canberra, and our initial possible world is still Buenos Aires. That is, this time we want to first pick a time table, and subsequently pick a noontime. This is beyond what can be expressed given the machinery as outlined previously. We have allowed predicate abstracts like $\langle \lambda y. \Phi(y) \rangle (f(x_1, \dots, x_n))$. In this each x_i is a variable, and so has some object as its value, as given by the valuation function of a model. Then, using the interpretation of f , the expression $f(x_1, \dots, x_n)$ designates an object too. The value of y then is, simply, some object in the domain of the model in which we are evaluating things. What we need is a higher type of variable that can take as its value a *function*, and not just an object. We would need to be able to say something like $\langle \lambda y. \Phi(y) \rangle (f)$, where y takes functions as values, and not objects. Returning to our concrete tide table example, picking the tide table first and the noontime second would become the following.

$$\langle \lambda x. \Diamond \langle \lambda y. P(x(y)) \rangle (c) \rangle (f)$$

If we had such machinery, it still would not be the end of the matter. The property represented by P is determined last—at Canberra in our evaluation procedure. It is quite plausible that we might want P to be evaluated first. That is, we might want to determine the meaning of P in Buenos Aires, and see if it applies to the time of the first high tide in Buenos Aires, following noon at Canberra. Again, this is something we can't say. We would need predicate abstraction over properties too. Something like the following.

$$\langle \lambda p. \langle \lambda x. \Diamond \langle \lambda y. p(x(y)) \rangle (c) \rangle (f) \rangle (P)$$

This is discussed at some length in the following section.

14.8 Extending The Notation to Predicates

We have seen that singular expressions—especially definite descriptions, which will be covered later—can vary in their extension from world to world. But if the definite description picks out different objects in different worlds, the description itself must differ in its extension in the different worlds. So, to use a familiar example, the number of planets need not have been eight: in different possible worlds different numbers fill that bill. But just as the extension of the singular term “the number of planets” varies from world to world, so too does the extension of the predicate expression “number of planets” out of which it is fashioned. This likewise varies in its extension from world to world. And sometimes it is its extension in this world that needs to enter the evaluation process in another possible world.

Examples are not easy to come by, but we have already seen one that arose in an unexpected context. It was put forward by Crossley and Humberstone (1977) as something that could not be handled as a scope issue. We gave it as (13.33), but we repeat it here for convenience.

It is possible for every red thing to be shiny. (14.10)

Here is the problem. Let $R(x)$ be x is red; and let $S(x)$ be x is shiny. As a first stab at symbolizing (14.10), we might try to read it as the *de dicto* claim

$$\Diamond(\forall x)(R(x) \supset S(x)). \quad (14.11)$$

Equation (14.11) says that there is a possible world w_i such that every thing that is red *in that world* w_i is shiny *in that world* w_i . But our original example (14.10) speaks about those things that are red *in this world*: it says that there is a possible world w_i in which these are all shiny *in that world* w_i . So, for a second stab, we try a *de re* reading,

$$(\forall x)(R(x) \supset \Diamond S(x)). \quad (14.12)$$

Informally, (14.12) says,

Each red thing is such that it is possible it is shiny. (14.13)

Equation (14.13) speaks about all the red things *in this world*, and it says with respect to each that there is a possible world in which it is shiny. But in (14.13) they need not all be shiny in the same world. In the original example (14.10), however, we want the actual red things all to be shiny in the same possible world. Equation (14.10) appears to be neither *de dicto* nor *de re*, and so, it has been argued, scope is insufficient to capture its logical form. (See also Humberstone 1981)

But scope considerations do play a role in understanding (14.13): it's not the scope of the subject expression that is at issue in this case but the scope of the predicate. It is all the red things that there are—the actual red things—that will, in another possible world, be shiny. We want to pick out that extension—the entirety of those individuals—in the actual world and move with it to some possible world. The notation we introduced in Sect. 14.7 using predicate variables should help. We can symbolize (14.13) as

$$(\forall x)(\lambda p.\Diamond(p(x) \supset S(x)))(R)(x). \quad (14.14)$$

Here is how we read (14.14):

Each object as well as the property *red* are such that there is a
possible world in which all of those red things in the former world
are shiny in the latter world. (14.15)

(14.15) is a mouthful, but it captures the intended meaning of (14.13).

Exercise 14.8.1 Symbolize this sentence: “There is a possible circumstance in which everyone who passed the Bar had not done so.”

References

- Bressan, A. (1972). *A general interpreted modal calculus*. Yale University Press.
- Crossley, J. N. & Humberstone, I. L. (1977). The logic of “actually”. *Reports on Mathematical Logic*, 8, 11–29.
- Fitting, M. (1972). An epsilon-calculus system for first-order S4. In W. Hodges (Ed.), *Conference in mathematical logic, london '70* (pp. 103–110). Springer Lecture Notes in Mathematics (Vol. 255). Springer.
- Fitting, M. (1973). A modal logic analog of Smullyan’s fundamental theorem. *Zeitschrift für mathematische Logik und Grundlagen der Mathematik*, 19, 1–16.
- Fitting, M. (1975). A modal logic epsilon-calculus. *Notre Dame Journal of Formal Logic*, 16, 1–16.
- Fitting, M. (1991). Modal logic should say more than it does. In J.-L. Lassez & G. Plotkin (Eds.), *Computational logic, essays in honor of Alan Robinson* (pp. 113–135). MIT Press.
- Fitting, M. (1993). Basic modal logic. In D. M. Gabbay, C. J. Hogger, & J. A. Robinson (Eds.), *Handbook of logic in artificial intelligence and logic programming* (Vol. 1, pp. 368–448). Oxford University Press.
- Fitting, M. (1996). A modal Herbrand theorem. *Fundamenta Informaticae*, 28, 101–122.
- Fitting, M. (1998). Higher-order modal logic—a sketch. In R. Caferra & G. Salzer (Eds.), *Automated deduction in classical and non-classical logics* (pp. 23–38). Springer Lecture Notes in Artificial Intelligence (Vol. 1761). Springer.
- Fitting, M. (2002a). Modal logics between propositional and first-order. *Journal of Logic and Computation*, 12, 1017–1026.
- Fitting, M. (2002b). *Types, tableaux, and Gödel’s god*. Errata at <http://melvinfitting.org/errata/errata.html>. Kluwer.
- Fitting, M. (2003). Intensional logic—beyond first order. In V. F. Hendricks & J. Malinowski (Eds.), *Trends in logic: 50 years of studia logica* (pp. 87–108). Kluwer Academic Publishers.
- Fitting, M. (2004). First-order intensional logic. *Annals of Pure and Applied Logic*, 127, 171–193.
- Fitting, M. (2006). FOIL axiomatized. *Studia Logica*, 84(1), 1–22. See Correction, Fitting, 2007.
- Fitting, M. (2007). Correction to FOIL Axiomatized. *Studia Logica*, 85(2), 275.
- Fitting, M. (2017). Rohit Parikh on logic, language and society. In R. R. C. Başkent, & L. Moss (Eds.), *Outstanding contributions to logic*. (Chap. 13 On Height and Happiness, Vol. 11, pp. 235–258). Springer.
- Humberstone, L. (1981). Scope and subjunctivity. *Philosophia*, 12, 99–126.
- Stalnaker, R. & Thomason, R. (1968). Abstraction in first-order modal logic. *Theoria*, 34, 203–207.
- Thomason, R. & Stalnaker, R. (1968). Modality and reference. *Nous*, 2, 359–372.
- Ward (Gunnvör), C. (2022). The viking answer lady. Click on “Daily Life”, then “How tall were the Vikings?” <http://www.vikinganswerlady.com>.

Chapter 15

Tableaus for Predicate Abstraction



15.1 Quantification and Non-rigidity

There are several propositional modal logics that we have looked at. The full range is infinite, and we only presented those of general interest for philosophical matters. Once quantifiers come in, the specific choice of propositional logic does not make a great difference, and most of our examples have simply assumed the properties of **K**, which made the discussion broadly applicable. At the level of quantification we have varying or constant domains. Either way, once predicate abstraction is brought in non-rigid constant and function symbols might always designate, or there may be possible worlds in which they do not designate. This gives us four different versions of quantified modal logics with predicate abstraction incorporated into the mechanism. We have seen these four versions semantically, and now we look at tableaus for them. We first introduce some convenient notation to designate the four possibilities.

Definition 15.1.1 (Notation Convention) For each propositional modal logic there are four quantified modal logics incorporating predicate abstraction that we can build on it. Here are the shorthand names we use to refer to them.

Constant Domain	{	terms always designate	CA
		terms might not designate	CN
Varying Domain	{	terms always designate	VA
		terms might not designate	VN

We will see that much of the difference between varying and constant domains comes down to a simple fact when non-rigidity is involved. If we use varying domains, the value of a non-rigid term at a world might be something that does not exist at that world, and so quantifiers do not include it in the range they are

quantifying over at that world. With constant domains, the value of a non-rigid term at a world must be something that exists there, and indeed everywhere. Of course in addition to this difference, constant and function symbols might always designate, or they might not designate at some worlds. This is the other key item from which the forms of our tableau rules arise.

The order of presentation in this chapter is the following. We begin with rules common to all four versions of tableaus. Next we discuss logics with constant quantificational domains, and finally those allowing varying domains. For both constant and varying domains, we first present the always-designate case, and then the might-not-designate case. The order of presentation is, generally speaking, from simpler to more complex throughout.

15.2 Object Terms Syntactically

What we say in this section applies to both constant and varying domain versions of our modal language. When first-order tableau rules were introduced in Chap. 9 the language used for proofs was extended with *parameters*, and these might have subscripts for the varying domain version, or omit them for the constant domain version. Parameters are not part of our basic modal language, but are part of a richer language used just for constructing tableau proofs. One can think of a tableau construction as a search for a counter-model to the formula we are trying to prove, and we can think of a prefix σ in a modal tableau as a candidate for a possible world. Then a varying domain parameter subscripted with that prefix, p_σ , can be thought of as being an object that is in the domain of the world σ , an existent at σ .

We are about to see yet another extension of the language used in proofs, needed because our formal modal language has been extended in Chap. 14 to include non-rigid constant and function symbols. What we need, loosely speaking, is some way of naming the objects that these non-rigid symbols denote at possible worlds when they do, in fact, denote. We present this machinery syntactically in the present section, and there is an informal semantics we use when talking about it. There will be a formal semantics for it in Sect. 16.1.

As we will see, constant and function symbols will acquire annotations in the course of tableau proof construction, somewhat like the way varying domain parameters have annotations, but the annotation has a different purpose now. A constant symbol c is allowed to be non-rigid. Continuing to think of prefixes in a tableau construction as playing the role of possible worlds, we intend non-rigid c , annotated with the prefix σ , to represent the object that c designates at possible world σ . This is a different role than the one played by the subscript on a varying domain parameter, which represents a possible world in which the parameter *exists*. An annotated constant symbol is intended to represent what the constant symbol *designates* at a world, though it may or may not exist at that world.

To keep notation clear we will not annotate constant symbols using *subscripts*, since the role of annotation for constant symbols is different than the one that

subscripts play for parameters. Instead we use *superscripts*, as in c^σ . Emphasizing the main point again, and speaking informally: the subscript on a varying domain parameter represents a possible world in which the parameter exists, while a superscripted constant symbol is a representation of what the constant symbol designates at a world, though it may or may not exist at that world. Similar remarks apply to function symbols, which also may acquire superscripts during a tableau construction.¹

Formally, in tableau proofs we may see constant symbols, c , d , ..., and superscripted versions, c^σ , c^τ , d^σ , d^τ , And similarly for function symbols. Informally, a constant symbol c is *non-rigid* and can have different values at different possible worlds, that is, at different prefixes. On the other hand, c^σ is meant to represent the value that c has in the possible world σ , and this is *rigid*. If c has the value c^σ at σ , it is true at every possible world that this is the value that c has at σ . Similarly for function symbols. Then scope distinctions are not relevant for superscripted constant and function symbols. A consequence of this is that superscripted function symbols can be directly nested, just as one sees in mathematics notation. We introduce a name, *object term*, for such constructions.

Definition 15.2.1 (Object Term) A variable is an object term. A parameter is an object term. If c is a constant symbol, c^σ is an object term, where σ is a prefix. If f is an n -place function symbol, σ is a prefix, and $t_1, \dots, t_n$ are object terms, then $f^\sigma(t_1, \dots, t_n)$ is an object term. An object term is *pseudo-closed* if it contains no free variables other than parameters.

For example, assume f and g are two-place function symbols, b is a constant symbol, $p_{1.1}$ is a varying domain parameter, and x is a variable. Then the expression $f^{1.1}(x, g^{1.1.1}(p_{1.1}, b^{1.1.1}))$ is an object term, but not one that is pseudo-closed because of the occurrence of the variable x . (Note, the term *pseudo-closed* first appeared in Definition 9.1.4.)

First-order formulas contain variables, with free and bound occurrences. For tableau constructions, *parameters* were introduced, subscripted for varying domain systems, and not subscripted for constant domain systems. Parameters were allowed to occur in places free variables could occur (indeed, they were a special kind of variable), but they were never allowed to occur bound. The class of object terms extends that of variables and parameters. Syntactically, for the purpose of tableau proofs, the set of allowed formulas is expanded once more, as follows.

Formulas that appear in tableau proofs may contain object terms in positions where parameters were previously allowed.

For example $(\forall x)P(x, a)$ was an allowed formula in a tableau, where a is a parameter. Now $(\forall x)P(x, f^\sigma(x, c^\tau))$ is also an allowed formula, where f is a

¹ The use of subscripts and superscripts is a change from the first edition of this book. In that edition, subscript notation was used throughout. We feel that explicitly distinguishing differences in informal meaning via differences in notation is the better choice.

function symbol, c is a constant symbol, and σ and τ are prefixes. It is still the case that parameters are never quantified.

15.3 Being on a Branch

Informally speaking, we can think of the collection of prefixed formulas on a tableau branch as a possible way things could be. Again informally speaking, if c^σ occurs on a branch, where c is a constant symbol and σ is a prefix, we can understand c^σ to be what c designates at possible world σ . In particular, if c^σ occurs on a branch, c can be assumed to designate something at σ . Now, how do superscripted constant symbols get to be added to a branch? Well, we will have a tableau rule that applies if $\sigma \langle \lambda x. \Phi(x) \rangle (c)$ is on the branch, and that allows us to add $\sigma \Phi(c^\sigma)$. (Recall the conventions on substitution that were introduced in Definition 8.1.5.) The rule premise $\sigma \langle \lambda x. \Phi(x) \rangle (c)$, which we can take to be true since it is on the branch, can be true only if c designates at σ and thus we are justified in our introduction of c^σ . With this understanding we could formulate other tableau rules making use of the easily stated assumption that c designates at σ if c^σ appears on the branch, because the only way it could have been introduced carries an assumption of designation. Right? Well, not quite. There is a small complication that requires a small detour.

Suppose we have $\sigma \langle \lambda x. \Phi(x) \rangle (c)$ on a tableau branch and, using a yet to be stated Predicate Abstraction Rule, we add $\sigma \Phi(c^\sigma)$ to the branch. What happens if x does not actually happen to have a free occurrence in $\Phi(x)$? Nothing in our rules for syntax says this is impossible. Then even though we had $\sigma \langle \lambda x. \Phi(x) \rangle (c)$ on the branch, we don't really have an occurrence of c^σ that has been added by application of this predicate abstract rule. Nonetheless, the fact that $\sigma \langle \lambda x. \Phi(x) \rangle (c)$ is on the branch is enough to ensure that c^σ must designate. And this is all we really need in order to make use of c^σ in any subsequent formulation of pseudo-closed terms. Similar issues arise with function symbols as well, of course. All this leads us to the introduction of the following terminology.

Definition 15.3.1 (Positively and Negatively Generated) In a prefixed tableau:

1. For a non-rigid constant symbol c we say:
 - a. c^σ is *positively generated* on a tableau branch if the branch contains $\sigma \langle \lambda x. \Phi(x) \rangle (c)$ for some $\Phi(x)$;
 - b. c^σ is *negatively generated* on a tableau branch if the branch contains $\sigma \neg \langle \lambda x. \Phi(x) \rangle (c)$ for some $\Phi(x)$.
2. For a non-rigid n -place function symbol f we say:
 - a. $f^\sigma(t_1, \dots, t_n)$ is *positively generated* on a tableau branch if the branch contains $\sigma \langle \lambda x. \Phi(x) \rangle (f(t_1, \dots, t_n))$ for some $\Phi(x)$;
 - b. $f^\sigma(t_1, \dots, t_n)$ is *negatively generated* on a tableau branch if the branch contains $\sigma \neg \langle \lambda x. \Phi(x) \rangle (f(t_1, \dots, t_n))$ for some $\Phi(x)$.

We use *generated* as short for *either positively or negatively generated*.

The point is, c^σ or $f^\sigma(t_1, \dots, t_n)$ can be generated on a tableau branch without actually appearing on the branch, something that can happen because of vacuous predicate abstracts. We note, as a matter of general interest, that similar issues of interpretation and use arise with vacuous quantification, and here things are not as simple as one might expect. We refer any interested reader to Fitting ([forthcoming](#)) for more details.

15.4 Constant Domain Tableau Rules

We begin with material that applies whether we have terms that always designate, or terms that might not designate. After this section, we become more specific about designation assumptions. Our propositional modal rules will be those for logics in the Lesser Modal Cube, from Chap. 7. Quantifier rules will be the constant domain version that was presented in Chap. 9. We will be assuming the earlier equality rules from Sect. 11.5, but with their range of application somewhat modified.

First, the Substitutivity Rule, Definition 11.5.2, allowed the replacement of equals by equals, *for parameters*. We now want it for the more general class of *object terms*.

Definition 15.4.1 (Atomic Substitutivity Rules, Constant Domain) For any atomic formula $\Phi(x)$, and for pseudo-closed object terms t and u ,

$$\frac{\sigma(t = u) \quad \tau \Phi(t)}{\tau \Phi(u)}$$

Next we turn to the Reflexivity Rule which, at first thought, seems very simple: everything equals itself. It is not simple, because it requires us to be precise about what makes up *everything*. As we will see, this is easy for constant domains, and not so easy for varying domains. In fact, for constant domains our original version of the rule, Definition 11.5.1 turns out to be quite sufficient. We restate it here for convenience.

Definition 15.4.2 (Reflexivity Rule, Constant Domain) Let p be a parameter, and σ be a prefix already on the branch. We have the following rule.

$$\overline{\sigma(p = p)}$$

When we come to varying domain tableaus in the next section, since non-rigid terms are allowed to designate nonexistent objects at worlds of varying domain models, explicit existence assumptions can be expected to play a significant role. We introduced an existence formula in Definition 12.1.2, $E(x)$, which abbreviates $(\exists y)(y = x)$. It is true at a varying domain possible world just when the value of x

is an existent at that world. One might think it would not be useful for constant domain models, since it would be true of every x , but this is exactly why it is useful. Building on $\mathbf{E}$ we define a predicate abstract expressing that a non-rigid term designates something that exists. It will play a significant role starting with Definition 15.4.5.

Definition 15.4.3 (Existence Abstract) By $\mathbf{E}$ we mean the predicate abstract $\langle \lambda x. \mathbf{E}(x) \rangle$. Then for a constant symbol c , for example, $\mathbf{E}(c)$ expands to the formula $\langle \lambda x. \mathbf{E}(x) \rangle(c)$ or fully unabbreviated, $\langle \lambda x. (\exists y)(y = x) \rangle(c)$.

Now we present tableau rules for predicate abstraction assuming constant domains. We begin with the simpler case where we always have designation. We remind you of the notation from Definition 15.1.1.

15.4.1 Constant Domains Assuming Terms Always Designate, CA

If we have $\sigma \langle \lambda x. \Phi(x) \rangle(c)$ on a tableau branch where c is a constant symbol then, informally speaking, the predicate abstract is true at possible world σ . And if this is the case c must designate at σ , so c^σ is meaningful, and then adding $\sigma \Phi(c^\sigma)$ to the branch can be allowed. If we have a negated abstract on a tableau branch, say $\sigma \neg \langle \lambda x. \Phi(x) \rangle(c)$, informally either c does not designate at σ , or it does but what it designates does not have the property that $\langle \lambda x. \Phi(x) \rangle$ determines. Since we are presenting rules for the always-designates case, we can rule out the first alternative, and conclude that what c designates at σ does not have the property that $\langle \lambda x. \Phi(x) \rangle$ determines. Of course similar things apply to function symbols. This motivates the rules for Predicate Abstraction below.

Definition 15.4.4 (Predicate Abstraction Rules, CA) For any constant symbol c , n -place function symbol f , and pseudo-closed object terms $t_1, \dots, t_n$:

$$\begin{array}{c} \frac{\sigma \langle \lambda x. \Phi(x) \rangle(c)}{\sigma \Phi(c^\sigma)} \qquad \frac{\sigma \neg \langle \lambda x. \Phi(x) \rangle(c)}{\sigma \neg \Phi(c^\sigma)} \\[2ex] \frac{\sigma \langle \lambda x. \Phi(x) \rangle(f(t_1, \dots, t_n))}{\sigma \Phi(f^\sigma(t_1, \dots, t_n))} \qquad \frac{\sigma \neg \langle \lambda x. \Phi(x) \rangle(f(t_1, \dots, t_n))}{\sigma \neg \Phi(f^\sigma(t_1, \dots, t_n))} \end{array}$$

Predicate abstract notation can be difficult to read; however, the rules above for using it are rather simple and quite mechanical, while allowing considerable flexibility. Here are two examples that show the rules above being applied. Suppose we have a tableau branch containing

$$\sigma \langle \lambda x. \langle \lambda y. \neg \langle \lambda z. \neg \Phi(z) \rangle(f(y)) \rangle(g(x)) \rangle(c),$$

where f and g are one place function symbols and c is a constant symbol. We are allowed to successively apply tableau rules as follows.

$$\begin{array}{ll}
 \sigma \langle \lambda x. \langle \lambda y. \neg \langle \lambda z. \neg \Phi(z) \rangle (f(y)) \rangle (g(x)) \rangle (c) & 1. \\
 \sigma \langle \lambda y. \neg \langle \lambda z. \neg \Phi(z) \rangle (f(y)) \rangle (g(c^\sigma)) & 2. \\
 \sigma \neg \langle \lambda z. \neg \Phi(z) \rangle (f(g^\sigma(c^\sigma))) & 3. \\
 \sigma \neg \neg \Phi(f^\sigma(g^\sigma(c^\sigma))) & 4. \\
 \sigma \Phi(f^\sigma(g^\sigma(c^\sigma))) & 5.
 \end{array}$$

Line 5 involves the pseudo-closed (in fact, closed) object term $f^\sigma(g^\sigma(c^\sigma))$ which we could read as “ f at σ of g at σ of c at σ ”.

We could also have a similar nested predicate abstract construct but with modal operators involved, say the formula

$$\langle \lambda x. \Diamond \langle \lambda y. \langle \lambda z. \Diamond \Phi(z) \rangle (f(y)) \rangle (g(x)) \rangle (c).$$

In this case we would get the following instead.

$$\begin{array}{ll}
 \sigma & \langle \lambda x. \Diamond \langle \lambda y. \Diamond \langle \lambda z. \Diamond \Phi(z) \rangle (f(y)) \rangle (g(x)) \rangle (c) \quad 1. \\
 \sigma & \Diamond \langle \lambda y. \Diamond \langle \lambda z. \Diamond \Phi(z) \rangle (f(y)) \rangle (g(c^\sigma)) \quad 2. \\
 \sigma.1 & \langle \lambda y. \Diamond \langle \lambda z. \Diamond \Phi(z) \rangle (f(y)) \rangle (g(c^\sigma)) \quad 3. \\
 \sigma.1 & \Diamond \langle \lambda z. \Diamond \Phi(z) \rangle (f(g^{\sigma.1}(c^\sigma))) \quad 4. \\
 \sigma.1.1 & \langle \lambda z. \Diamond \Phi(z) \rangle (f(g^{\sigma.1}(c^\sigma))) \quad 5. \\
 \sigma.1.1 & \Diamond \Phi(f^{\sigma.1.1}(g^{\sigma.1}(c^\sigma))) \quad 6. \\
 \sigma.1.1.1 & \Phi(f^{\sigma.1.1}(g^{\sigma.1}(c^\sigma))) \quad 7.
 \end{array}$$

This time line 7 involves the pseudo-closed object term $f^{\sigma.1.1}(g^{\sigma.1}(c^\sigma))$, or “ f at $\sigma.1.1$ of g at $\sigma.1$ of c at σ ”. Line 7 informally asserts that Φ is true of this, at possible world $\sigma.1.1.1$.

With constant domains, terms that designate must designate *always existent* objects. This is the content of the next set of rules, which call on Definition 15.4.3.

Definition 15.4.5 (Existence Rules, CA) For any constant symbol c , n -place function symbol f , and pseudo-closed object terms $t_1, \dots, t_n$:

$$\begin{array}{ll}
 \frac{\sigma \langle \lambda x. \Phi(x) \rangle (c)}{\sigma \mathbf{E}(c)} & \frac{\sigma \neg \langle \lambda x. \Phi(x) \rangle (c)}{\sigma \mathbf{E}(c)} \\
 \\
 \frac{\sigma \langle \lambda x. \Phi(x) \rangle (f(t_1, \dots, t_n))}{\sigma \mathbf{E}(f(t_1, \dots, t_n))} & \frac{\sigma \neg \langle \lambda x. \Phi(x) \rangle (f(t_1, \dots, t_n))}{\sigma \mathbf{E}(f(t_1, \dots, t_n))}
 \end{array}$$

This completes the constant domain tableau rules, assuming terms always designate.

Example 15.4.6 Here is a **K** tableau proof using the *CA* rules, of the sentence $\Diamond\langle\lambda x.P(x)\rangle(c) \supset (\exists x)\Diamond P(x)$, where c is a constant symbol.

- 1 $\neg[\Diamond\langle\lambda x.P(x)\rangle(c) \supset (\exists x)\Diamond P(x)]$ 1.
- 1 $\Diamond\langle\lambda x.P(x)\rangle(c)$ 2.
- 1 $\neg(\exists x)\Diamond P(x)$ 3.
- 1.1 $\langle\lambda x.P(x)\rangle(c)$ 4.
- 1.1 $P(c^{1.1})$ 5.
- 1.1 **E**(c) 6.
- 1.1 $\langle\lambda x.(\exists y)(y = x)\rangle(c)$ 7.
- 1.1 $(\exists y)(y = c^{1.1})$ 8.
- 1.1 $p = c^{1.1}$ 9.
- 1 $\neg\Diamond P(p)$ 10.
- 1.1 $\neg P(p)$ 11.
- 1.1 $P(p)$ 12.

In this, 2 and 3 are from 1 by an Implication Rule; 4 is from 2 by a Possibility Rule; 5 is from 4 by a (positive) Abstraction Rule; 6 is from 4 by a (positive) Existence Rule, 7 is 6 unabbreviated, 8 is from 7 by a (positive) Abstraction Rule, 9 is from 8 by an Existential Quantifier Rule, 10 is from 3 by a Universal Quantifier Rule, 11 is from 10 by a Necessitation rule, and 12 is from 5 and 9 by an Atomic Substitutivity Rule. Closure is by 11 and 12.

A close look at the proof above shows that the Existence Rules, combined with the rules for the existential quantifier, tell us that a superscripted constant symbol is equal to a parameter. Then, via the substitutivity of equality, our universal quantifier rules, and our reflexivity rule for tableaus directly carry over to the superscripted constant symbol. Function symbols are a little more complicated, but not difficult to handle. All this can be incorporated into quite natural derived rules that can take care of much of the work for us.

Proposition 15.4.7 (Derived Parameter Introduction Rule, CA) *Assume we have a tableau branch for modal logic **L** using the CA assumptions. Let t be a psuedo-closed object term that is generated on the branch, Definition 15.3.1. Also let σ be a prefix occurring on the branch. Then the following is a sound derived rule for that branch, where p is any parameter new to the branch.*

$$\frac{}{\sigma(p = t)}$$

Proof There are two cases: t is a constant symbol or t involves a function symbol. We only give the constant symbol case; the other is similar. Suppose t is c^τ , which

is positively generated on branch $\mathcal{B}$ (negative generation is similar). And suppose σ appears on the branch. Then branch $\mathcal{B}$ must be like the following.

$$\begin{array}{l} \vdots \\ \sigma \Psi \quad 1. \\ \vdots \\ \tau \langle \lambda x. \Phi_1 \rangle (c) \quad 2. \\ \vdots \end{array}$$

Here line 1 is simply because prefix σ occurs on the branch, and line 2 is because c^τ is positively generated on the branch. We can continue the branch as follows.

$$\begin{array}{l} \vdots \\ \sigma \Psi \quad 1. \\ \vdots \\ \tau \langle \lambda x. \Phi_1 \rangle (c) \quad 2. \\ \vdots \\ \tau \mathbf{E}(c) \quad 3. \\ \tau \langle \lambda x. (\exists y)(y = x) \rangle (c) \quad 4. \\ \tau (\exists y)(y = c^\tau) \quad 5. \\ \tau p = c^\tau \quad 6. \\ \sigma p = p \quad 7. \\ \sigma p = c^\tau \quad 8. \end{array}$$

Line 3 is from line 2 using an Existence Rule. Line 4 is 3 unabbreviated. Line 5 is from line 4 by a Predicate Abstraction Rule. Line 6 is from line 5 by an Existential Quantifier Rule; in it p is an arbitrary new parameter. Line 7 is by Reflexivity, Definition 15.4.2. Finally line 8 is from lines 6 and 7 by an Atomic Substitutivity Rule, Definition 15.4.1.

Here are some easy consequences of this derived rule.

Proposition 15.4.8 (Derived Reflexivity Rule, CA) *Let t be a psuedo-closed object term that is generated on the branch, Definition 15.3.1, and let σ be a prefix occurring on the branch. Then the following is a sound derived rule for that branch.*

$$\frac{}{\sigma t = t}$$

Proof This is just a sketch. On the tableau branch, add $\sigma(p = t)$ using Proposition 15.4.7. Next add $\sigma(p = p)$ using the constant domain reflexivity rule, Definition 15.4.2. Finally, apply Atomic Substitutivity twice, Definition 15.4.1.

The following essentially says that as far as equality of object terms is concerned, all possible worlds behave alike.

Proposition 15.4.9 (Universality of Equality Derived Rule, CA) *For all pseudo-closed object terms t and u , if prefix τ already appears on a tableau branch then the following inference is sound.*

$$\frac{\sigma(t = u)}{\tau(t = u)}$$

Proof Assume $\sigma(t = u)$ occurs on a tableau branch. Since t appears on the branch, it will be the case that t meets the conditions required by the Derived Reflexivity Rule (as does u). Then we can proceed as follows.

$$\begin{array}{ll} \sigma(t = u) & 1. \\ \tau(t = t) & 2. \\ \tau(t = u) & 3. \end{array}$$

In this, 1 is an assumption. Since t meets the conditions for the Reflexivity rule, we have 2. Then 3 follows by (atomic) Substitutivity on 1 and 2 by taking $\Phi(x)$ to be $(t = x)$.

The universal quantification rules allow us to instantiate using any *parameter*. The following extends this to object terms in general.

Proposition 15.4.10 (Derived Universal Rules, CA) *For a given tableau branch let t be a pseudo-closed object term that is generated on the branch (Definition 15.3.1). Then the following are sound derived rules on that branch.*

$$\frac{\sigma(\forall x)\Phi(x)}{\sigma\Phi(t)} \qquad \frac{\sigma\neg(\exists x)\Phi(x)}{\sigma\neg\Phi(t)}$$

Proof Briefly sketched, the conditions here and Proposition 15.4.7 allow us to add $\sigma p = t$ to the branch, where p is a new parameter. Standard universal rules allow us to add $\sigma\Phi(p)$ (or $\sigma\neg\Phi(p)$), depending on which of the two rule versions we are considering). Now use the General Substitutivity version of Definition 15.4.1, which is a derived rule by an argument exactly like the one used for equality in Chapter 11, and which will be repeated here.

Example 15.4.11 Here is one more **K** tableau proof using the CA rules. It is presented in an abbreviated form, using the derived rules just established. We show the following $(\forall x)\Diamond(\lambda y.R(x, y))(f(x)) \supset (\forall x)\Diamond(\exists y)R(x, y)$,

- 1 $\neg[(\forall x)\Diamond\langle\lambda y.R(x, y)\rangle(f(x)) \supset (\forall x)\Diamond(\exists y)R(x, y)]$ 1.
- 1 $(\forall x)\Diamond\langle\lambda y.R(x, y)\rangle(f(x))$ 2.
- 1 $\neg(\forall x)\Diamond(\exists y)R(x, y)$ 3.
- 1 $\neg\Diamond(\exists y)R(p, y)$ 4.
- 1 $\Diamond\langle\lambda y.R(p, y)\rangle(f(p))$ 5.
- 1.1 $\langle\lambda y.R(p, y)\rangle(f(p))$ 6.
- 1.1 $\neg(\exists y)R(p, y)$ 7.
- 1.1 $R(p, f^{1.1}(p))$ 8.
- 1.1 $\neg R(p, f^{1.1}(p))$ 9.

In this, 2 and 3 are from 1 by an Implication Rule; 4 is from 3 by an Existential Rule; 5 is from 2 by a Universal Rule; 6 is from 5 by a Possibility Rule; 7 is from 4 by a Necessity Rule; 8 is from 6 by a (positive) Predicate Abstraction Rule; and 9 is from 7 by a Derived Universal Rule, [15.4.10](#).

15.4.2 Constant Domains Assuming Terms Might Not Designate, CN

If constant or function symbols might not designate, the negated abstract rules from the previous section are no longer always applicable. Side conditions for when they are applicable must be imposed. Things will be arranged so that the only way c^σ can be introduced to a tableau branch is under circumstances where it must designate because of the presence on the tableau branch of a positive predicate abstract of the form $\sigma \langle\lambda x.\Psi(x)\rangle(c)$. Without this, we don't know if c^σ designates on the branch, and if it happens that it doesn't we can't introduce it in a rule involving a negated abstract. Similarly for function symbols. This accounts for the side conditions below.

Definition 15.4.12 (Predicate Abstraction Rules, CN) For any constant symbol c , n -place function symbol f , and pseudo-closed object terms $t_1, \dots, t_n$:

$$\begin{array}{c}
 \frac{\sigma \langle\lambda x.\Phi(x)\rangle(c)}{\sigma \Phi(c^\sigma)} \\
 \\
 \frac{\sigma \langle\lambda x.\Phi(x)\rangle(f(t_1, \dots, t_n))}{\sigma \Phi(f^\sigma(t_1, \dots, t_n))} \\
 \\
 \frac{\sigma \neg\langle\lambda x.\Phi(x)\rangle(c)}{\sigma \neg\Phi(c^\sigma)} \\
 \text{provided } c^\sigma \text{ is positively} \\
 \text{generated on the tableau branch} \\
 \\
 \frac{\sigma \neg\langle\lambda x.\Phi(x)\rangle(f(t_1, \dots, t_n))}{\sigma \neg\Phi(f^\sigma(t_1, \dots, t_n))} \\
 \text{provided } f^\sigma(t_1, \dots, t_n) \text{ is positively} \\
 \text{generated on the tableau branch}
 \end{array}$$

The Existence Rules have a corresponding change eliminating the negative generation cases.

Definition 15.4.13 (Existence Rules, *CN*) For any constant symbol c , n -place function symbol f , and pseudo-closed object terms $t_1, \dots, t_n$:

$$\frac{\sigma \langle \lambda x. \Phi(x) \rangle (c)}{\sigma \mathbf{E}(c)} \qquad \frac{\sigma \langle \lambda x. \Phi(x) \rangle (f(t_1, \dots, t_n))}{\sigma \mathbf{E}(f(t_1, \dots, t_n))}$$

This completes the constant domain tableau rules, assuming terms might not designate.

The derived rules in Propositions 15.4.7, 15.4.8, 15.4.9, and 15.4.10 have the obvious modification: “generated” should be replaced with “positively generated” in both the statements and their proofs. We do not bother to restate things. We leave it to you to check that the tableaus in Examples 15.4.6 and 15.4.11 are still correct, even if terms might not designate.

Exercises

Exercise 15.4.1 Give a **K** tableau proof of the following sentence, using the *CN* assumptions.

$$\langle \lambda x. \Diamond P(x) \rangle (c) \supset (\exists x) \Diamond P(x)$$

where c is a constant symbol.

Exercise 15.4.2 Give a **K** proof of the following formula, using *CN* conditions.

$$\Box(\forall x)[A(x) \supset \langle \lambda y. B(y) \rangle (f(x))] \supset [\Diamond \langle \lambda x. A(x) \rangle (c) \supset \Diamond(\exists x)B(x)]$$

(You should compare this with Exercise 15.5.6.)

Exercise 15.4.3 Give a constant domain proof in **K** of $A \supset C$ using the *CN* conditions, where A and C are defined as follows.

$$A = (\forall x) \Diamond \langle \lambda y. (\forall z) \Box \langle \lambda w. R(x, y, z, w) \rangle (g(x, z)) \rangle (f(x))$$

$$C = (\forall x) \Diamond (\exists y) (\forall z) \Box (\exists w) R(x, y, z, w)$$

(Compare this with Exercise 15.5.7.)

15.5 Varying Domain Tableau Rules

When varying domains are involved, we can no longer assume that an object that exists somewhere exists everywhere. As a consequence when we encounter an object term like c^σ , we can still understand it as what the non-rigid constant symbol

c designates at σ , but we don't know where that object 'lives'. In particular, it need not exist at σ . Our constant domain Existence Rules, Definition 15.4.5, are no longer appropriate. This requires us to strengthen the Reflexivity Rule from the constant domain version. There we assumed reflexivity just for parameters, and derived a more general version, Proposition 15.4.8. Here we will simply take the counterpart of the derived version as primitive. We assume the usual propositional tableau and modal rules. Quantifier rules are those for varying domains, from Sect. 9.2. As we did for constant domains, we begin with the case in which terms always designate, then we modify the rules to adapt them to the possibility of non-designation.

15.5.1 Varying Domains, Assuming Terms Always Designate, VA

To begin, the rules for predicate abstractions are exactly as in the CA version, except that now parameters are subscripted. We restate the rules for convenience.

Definition 15.5.1 (Predicate Abstraction Rules, VA) For any constant symbol c , n -place function symbol f , and pseudo-closed object terms $t_1, \dots, t_n$:

$$\frac{\sigma \langle \lambda x. \Phi(x) \rangle (c)}{\sigma \Phi(c^\sigma)} \qquad \frac{\sigma \neg \langle \lambda x. \Phi(x) \rangle (c)}{\sigma \neg \Phi(c^\sigma)}$$

$$\frac{\sigma \langle \lambda x. \Phi(x) \rangle (f(t_1, \dots, t_n))}{\sigma \Phi(f^\sigma(t_1, \dots, t_n))} \qquad \frac{\sigma \neg \langle \lambda x. \Phi(x) \rangle (f(t_1, \dots, t_n))}{\sigma \neg \Phi(f^\sigma(t_1, \dots, t_n))}$$

Here is an example making use of these rules.

Example 15.5.2 (Assuming VA) Exercise 14.2.2 asked for semantic verification of several formulas involving predicate abstraction and constant symbols, under the assumption that constant symbols always designated. We look at one of these, coming from part 1 of the exercise. Here is a tableau proof of $\neg \langle \lambda x. \Phi(x) \rangle (c) \supset \langle \lambda x. \neg \Phi(x) \rangle (c)$.

- 1 $\neg[\neg \langle \lambda x. \Phi(x) \rangle (c) \supset \langle \lambda x. \neg \Phi(x) \rangle (c)]$ 1.
- 1 $\neg \langle \lambda x. \Phi(x) \rangle (c)$ 2.
- 1 $\neg \langle \lambda x. \neg \Phi(x) \rangle (c)$ 3.
- 1 $\neg \Phi(c^1)$ 4.
- 1 $\neg \neg \Phi(c^1)$ 5.

Here 2 and 3 are from 1 by a Conjunctive Rule. Then 4 is from 2, and 5 is from 3 by Predicate Abstraction rules. Closure is by 4 and 5.

For equality rules, substitutivity remains as it was in the constant domain case (except that object terms now must have subscripted parameters). We restate the rule for convenience.

Definition 15.5.3 (Atomic Substitutivity Rules, VA) For any atomic formula $\Phi(x)$, and for pseudo-closed object terms t and u ,

$$\frac{\sigma(t = u) \quad \tau \Phi(t)}{\tau \Phi(u)}$$

Our reflexivity rule cannot now be like the constant domain version, Definition 15.4.2, which involved only a parameter. Instead we take as a primitive rule a version of what was a derived rule for constant domains (see Proposition 15.4.8).

Definition 15.5.4 (Reflexivity Rule, VA) Let t be a parameter or a pseudo-closed object term that is generated on the branch, Definition 15.3.1, and let σ be a prefix occurring on the branch. We have the following tableau rule.

$$\frac{}{\sigma t = t}$$

Here is an example that makes use of equality.

Example 15.5.5 (Assuming VA) We provide a **K** proof of the following somewhat mysterious formula, where a and p are non-rigid constant symbols. In Example 17.2.1 we will provide a scenario in which this has a natural meaning. Until then, just consider it a formal exercise.

$$\left[\langle \lambda x. \Diamond \langle \lambda y. x = y \rangle(p) \rangle(a) \wedge \langle \lambda x. \Box S(x) \rangle(a) \right] \supset \Diamond \langle \lambda x. S(x) \rangle(p).$$

Here is a proof, assuming the **K** rules with the VA conditions.

- 1 $\neg \{ [\langle \lambda x. \Diamond \langle \lambda y. x = y \rangle(p) \rangle(a) \wedge \langle \lambda x. \Box S(x) \rangle(a)] \supset \Diamond \langle \lambda x. S(x) \rangle(p) \}$ 1.
- 1 $\langle \lambda x. \Diamond \langle \lambda y. x = y \rangle(p) \rangle(a) \wedge \langle \lambda x. \Box S(x) \rangle(a)$ 2.
- 1 $\neg \Diamond \langle \lambda x. S(x) \rangle(p)$ 3.
- 1 $\langle \lambda x. \Diamond \langle \lambda y. x = y \rangle(p) \rangle(a)$ 4.
- 1 $\langle \lambda x. \Box S(x) \rangle(a)$ 5.
- 1 $\Diamond \langle \lambda y. a^1 = y \rangle(p)$ 6.
- 1.1 $\langle \lambda y. a^1 = y \rangle(p)$ 7.
- 1.1 $a^1 = p^{1.1}$ 8.
- 1.1 $\neg \langle \lambda x. S(x) \rangle(p)$ 9.
- 1.1 $\neg S(p^{1.1})$ 10.
- 1 $\Box S(a^1)$ 11.
- 1.1 $S(a^1)$ 12.
- 1.1 $S(p^{1.1})$ 13.

In this, 2 and 3 are from 1 by an Implication Rule, and 4 and 5 are from 2 by a Conjunctive Rule; 6 is from 4 by a Positive Abstraction Rule; 7 is from 6 by a Possibility Rule; 8 is from 7 by Positive Abstraction; 9 is from 3 by a Necessity Rule; 10 is from 9 by Negative Abstraction; 11 is from 5 by Positive Abstraction; 12 is from 11 by a Necessity Rule; and 13 is from 8 and 12 by (atomic) Substitutivity. Closure is by 10 and 13.

An important fact about our quantifier rules is that they only involve *parameters*. More general object terms play no direct role. For constant domain tableau rules we had Existence Rules, 15.4.5 and 15.4.13, and these served to connect object terms with parameters. We cannot adopt those rules now because with varying domains, a constant symbol c might designate at a possible world (prefix) σ but it need not designate an existent at σ . However, if we have explicit information about existence, things are different.

Example 15.5.6 (Assuming VA) The formula $(\forall x)P(x) \supset \langle \lambda x.P(x) \rangle(c)$ is not valid, where c is a constant symbol. At a particular world c might designate an object that does not exist there, and so is not in the range of the universal quantifier. But here is a tableau proof of $[(\forall x)P(x) \wedge \mathbf{E}(c)] \supset \langle \lambda x.P(x) \rangle(c)$. As it happens, modal issues are not involved in this example.

$$\begin{array}{ll}
 1 \neg\{[(\forall x)P(x) \wedge \mathbf{E}(c)] \supset \langle \lambda x.P(x) \rangle(c)\} & 1. \\
 1 \quad (\forall x)P(x) \wedge \mathbf{E}(c) & 2. \\
 1 \neg\langle \lambda x.P(x) \rangle(c) & 3. \\
 1 \quad (\forall x)P(x) & 4. \\
 1 \quad \mathbf{E}(c) & 5. \\
 1 \quad \langle \lambda x.(\exists y)(y = x) \rangle(c) & 6. \\
 1 \quad (\exists y)(y = c^1) & 7. \\
 1 \quad p_1 = c^1 & 8. \\
 1 \quad P(p_1) & 9. \\
 1 \quad P(c^1) & 10. \\
 1 \neg P(c^1) & 11.
 \end{array}$$

In this, 2 and 3 are from 1 by an Implication Rule, and 4 and 5 are from 2 by a Conjunctive Rule. Then 6 is 5 unabbreviated; 7 is from 6 by a Positive Abstraction Rule; 8 is from 7 by an Existential Rule (where p_1 is a new parameter); 9 is from 4 by a Universal Rule; 10 is from 8 and 9 by (atomic) Substitutivity; 11 is from 3 by a Negative Abstraction Rule.

The following generalizes what went on in the example we just presented. Its proof is left as an exercise.

Proposition 15.5.7 (Object Existence Derived Rule, VA) *Assuming General substitutivity of equality (not just Atomic), for any formula $\Phi(x)$, constant symbol c , n -ary function symbol f , and pseudo-closed object terms $t_1, \dots, t_n$:*

$$\frac{\sigma \mathbf{E}(c)}{\frac{\sigma (\forall x)\Phi(x)}{\sigma \Phi(c^\sigma)}} \qquad \frac{\sigma \mathbf{E}(c)}{\frac{\sigma \neg(\exists x)\Phi(x)}{\sigma \neg\Phi(c^\sigma)}}$$

$$\frac{\sigma \mathbf{E}(f(t_1, \dots, t_n))}{\frac{\sigma (\forall x)\Phi(x)}{\sigma \Phi(f^\sigma(t_1, \dots, t_n))}} \qquad \frac{\sigma \mathbf{E}(f(t_1, \dots, t_n))}{\frac{\sigma \neg(\exists x)\Phi(x)}{\sigma \neg\Phi(f^\sigma(t_1, \dots, t_n))}}$$

Here is a final and somewhat more complex example. Note that the antecedent makes an explicit assumption about the existence of function values for the function f .

Example 15.5.8 (Assuming VA) The following is a **K** tableau proof, using General Substitutivity, of

$$\{(\forall x)\Box\mathbf{E}(f(x)) \wedge (\forall x)\Diamond\langle\lambda y.R(x, y)\rangle(f(x))\} \supset (\forall x)\Diamond(\exists y)R(x, y)$$

- 1 $\neg [\{(\forall x)\Box\mathbf{E}(f(x)) \wedge (\forall x)\Diamond\langle\lambda y.R(x, y)\rangle(f(x))\} \supset (\forall x)\Diamond(\exists y)R(x, y)]$ 1.
- 1 $(\forall x)\Box\mathbf{E}(f(x)) \wedge (\forall x)\Diamond\langle\lambda y.R(x, y)\rangle(f(x))$ 2.
- 1 $\neg(\forall x)\Diamond(\exists y)R(x, y)$ 3.
- 1 $(\forall x)\Box\mathbf{E}(f(x))$ 4.
- 1 $(\forall x)\Diamond\langle\lambda y.R(x, y)\rangle(f(x))$ 5.
- 1 $\neg\Diamond(\exists y)R(p_1, y)$ 6.
- 1 $\Diamond\langle\lambda y.R(p_1, y)\rangle(f(p_1))$ 7.
- 1.1 $\langle\lambda y.R(p_1, y)\rangle(f(p_1))$ 8.
- 1.1 $R(p_1, f^{1.1}(p_1))$ 9.
- 1.1 $\neg(\exists y)(R(p_1, y))$ 10.
- 1 $\Box\mathbf{E}(f(p_1))$ 11.
- 1.1 $\mathbf{E}(f(p_1))$ 12.
- 1.1 $\neg R(p_1, f^{1.1}(p_1))$ 13.

Items 2 and 3 are from 1 by an Implication Rule; 4 and 5 are from 2 by a Conjunctive Rule; 6 is from 3 by an Existential Rule; 7 is from 5 by a Universal Rule; 8 is from 7 by a Possibility Rule; 9 is from 8 by a Predicate Abstraction Rule; 10 is from 6 by a Necessitation Rule; 11 is from 4 by a Universal Rule.; 12 is from 11 by a Necessitation Rule; 13 is from 10 and 12 using the Object Existence Derived Rule. Now this branch closes because of 9 and 13.

15.5.2 *Varying Domains, Assuming Terms Might Not Designate, VN*

Dropping from VA the requirement that constant and function symbols always designate has much the same effects that it did in the constant domain case with CA. Much stays the same. The quantifier rules remain unchanged, as do the propositional, modal, and substitutivity rules. Differences appear only in the predicate abstraction rules themselves, and in the reflexivity rule for equality, and the differences are entirely in the side conditions specifying when a rule is applicable.

Definition 15.5.9 (Predicate Abstraction Rules, VN) For any constant symbol c , n -place function symbol f , and pseudo-closed object terms $t_1, \dots, t_n$:

$$\begin{array}{c}
 \frac{\sigma \langle \lambda x. \Phi(x) \rangle (c)}{\sigma \Phi(c^\sigma)} \\
 \\
 \frac{\sigma \neg \langle \lambda x. \Phi(x) \rangle (c)}{\sigma \neg \Phi(c^\sigma)} \\
 \text{provided } c^\sigma \text{ is positively} \\
 \text{generated on the tableau branch} \\
 \\
 \frac{\sigma \langle \lambda x. \Phi(x) \rangle (f(t_1, \dots, t_n))}{\sigma \Phi(f^\sigma(t_1, \dots, t_n))} \\
 \\
 \frac{\sigma \neg \langle \lambda x. \Phi(x) \rangle (f(t_1, \dots, t_n))}{\sigma \neg \Phi(f^\sigma(t_1, \dots, t_n))} \\
 \text{provided } f^\sigma(t_1, \dots, t_n) \text{ is positively} \\
 \text{generated on the tableau branch}
 \end{array}$$

Using these rules, Example 15.5.2 no longer goes through. In the tableau construction, no tableau rule applies after step 3 if we use the rule versions from Definition 15.5.9. That is, $\neg \langle \lambda x. \Phi(x) \rangle (c) \supset \langle \lambda x. \neg \Phi(x) \rangle (c)$ is not provable if non-designation for constant symbols is allowed. On the other hand the converse, $\langle \lambda x. \neg \Phi(x) \rangle (c) \supset \neg \langle \lambda x. \Phi(x) \rangle (c)$, remains provable, as you can check.

While the Substitutivity rule for Equality, Definition 15.5.3, does not change from the always-designates case, the Reflexivity Rule does require modification. We have already noted that the Reflexivity Rule is complex because it requires us to know what a ‘thing’ is. When we were dealing with terms that always designated, this was a rather minor problem. If σ was a prefix that had been previously used on a tableau branch, then informally it represents a possible world that we know exists. And then since function symbols always designated, for any function symbol g we have that g^σ must be meaningful, whether introduced positively or through a negation. We no longer have that simplicity, since a function symbol or a constant symbol need not designate now. We want some way of determining when we have one that does, and once again, we make use of the terminology introduced in Definition 15.3.1.

Definition 15.5.10 (Reflexivity Rule, VN) Let t be a parameter or a pseudo-closed object term that is *positively* generated on the branch, Definition 15.3.1. Then the following inference is allowed provided the prefix σ is already on the branch.

$$\frac{}{\sigma(t = t)}$$

It is easy to check that the Universality of Equality Derived Rule, Proposition 15.4.9, still holds. The Object Existence Derived Rule, Proposition 15.5.7, continues to hold. Its proof is left as an exercise, as are the verifications that Examples 15.5.5, 15.5.6 and 15.5.8 are still correct.

Exercises

Exercise 15.5.1 For each of the four tableau versions, *CA*, *CN*, *VA*, and *VN* from Definition 15.1.1, either give a **K** tableau proof or give a **K** countermodel for each of the following formulas .

1. $(\forall x)\Phi(x) \supset \langle \lambda x. \Phi(x) \rangle(c)$
2. $(\forall x)\Phi(x) \supset \neg \langle \lambda x. \neg \Phi(x) \rangle(c)$
3. $\langle \lambda x. \Phi(x) \rangle(c) \supset \neg \langle \lambda x. \neg \Phi(x) \rangle(c)$
4. $\neg \langle \lambda x. \neg \Phi(x) \rangle(c) \supset \langle \lambda x. \Phi(x) \rangle(c)$

Exercise 15.5.2 Let a and b be non-rigid constant symbols. Try giving a varying domain **K** tableau proof of $\langle \lambda x, y. \Diamond(x = y) \rangle(a, b) \supset \langle \lambda x, y. (x = y) \rangle(a, b)$, under both the *VN* and the *VA* assumptions. (We have made use of the abbreviated notation from Definition 14.1.3.)

Exercise 15.5.3 Discuss what goes wrong with an attempt to provide a varying domain **K** proof of $\langle \lambda x, y. (x = y) \rangle(a, b) \supset \Box \langle \lambda x, y. (x = y) \rangle(a, b)$. Assume the constant symbols a and b always designate, *VN*. Contrast this with the provability of $\langle \lambda x, y. (x = y) \rangle(a, b) \supset \langle \lambda x, y. \Box(x = y) \rangle(a, b)$, and discuss the difference informally. Again we have made use of the abbreviated notation from Definition 14.1.3.

Exercise 15.5.4 Prove Proposition 15.5.7, the Object Existence Derived Rule. Make sure your proof works even if terms are allowed to not designate.

Exercise 15.5.5 Verify that the following Examples still work, even if terms might not designate.

1. Example 15.5.5
2. Example 15.5.6
3. Example 15.5.8

Exercise 15.5.6 For varying domain **K**, assuming that terms might not designate, *VN*:

1. Give a tableau proof of this formula.

$$\{\Box(\forall x)[A(x) \supset \langle \lambda y.B(y) \rangle(f(x))]\} \wedge \Box(\forall x)\mathbf{E}(f(x)) \supset \\ \{\Diamond[\langle \lambda x.A(x) \rangle(c) \wedge \mathbf{E}(c)] \supset \Diamond(\exists x)B(x)\}$$

2. Show that the following formula is not valid in some varying domain first-order **K** model in which terms might not designate.

$$\{\Box(\forall x)[A(x) \supset \langle \lambda y.B(y) \rangle(f(x))]\} \wedge \Box(\forall x)\mathbf{E}(f(x)) \supset \\ \{\Diamond\langle \lambda x.A(x) \rangle(c) \supset \Diamond(\exists x)B(x)\}$$

Exercise 15.5.7 Assuming the *VN* rules, give a **K** tableau proof of $(A \wedge B) \supset C$, where A , B , and C are defined as follows.

$$\begin{aligned} A &= (\forall x)\Diamond\langle \lambda y.(\forall z)\Box\langle \lambda w.R(x, y, z, w) \rangle(g(x, z)) \rangle(f(x)) \\ B &= (\forall x)\Box[\mathbf{E}(f(x)) \wedge (\forall y)\Box\mathbf{E}(g(x, y))] \\ C &= (\forall x)\Diamond(\exists y)(\forall z)\Box(\exists w)R(x, y, z, w) \end{aligned}$$

(Compare this with Exercise 15.4.3.)

Exercise 15.5.8 This concerns formula (19.10) from Sect. 19.4, $(\forall x)[\langle \lambda y.(y = x) \rangle(c) \supset P(x)] \equiv \langle \lambda x.P(x) \rangle(c)$. Give **K** tableau proofs, assuming *VN*, of the following.

1. $\mathbf{E}(c) \supset \{(\forall x)[\langle \lambda y.(y = x) \rangle(c) \supset P(x)] \supset \langle \lambda x.P(x) \rangle(c)\}$
2. $\langle \lambda x.P(x) \rangle(c) \supset (\forall x)[\langle \lambda y.(y = x) \rangle(c) \supset P(x)]$

Exercise 15.5.9 This will be used to complete an argument in Sect. 19.4. Give a varying domain **D** proof that the conjunction of items 1, 2, and 3 below implies item 4.

1. $\mathbf{E}(c)$ (using Definition 15.4.3)
2. $(\forall x)\Box\neg(x = x + 1)$
3. $\langle \lambda x.\Box\langle \lambda y.(y = x + 1) \rangle(c) \rangle(c)$
4. $\neg\langle \lambda x.\Box\langle \lambda y.y = x \rangle(c) \rangle(c)$

Reference

Fitting, M. (forthcoming). Walter Carnielli on reasoning, paraconsistency, and probability. In H. Antunes, A. Freire, & A. Rodrigues (Eds.), (Chap. A logic not exactly adopted). Outstanding contributions to logic. Springer.

Chapter 16

Tableau Soundness and Completeness



We have seen several soundness and completeness proofs so far. And we have just introduced four more tableau systems in need of such proofs. For each of the modal logics from the Lesser Modal Cube, Fig. 7.1, we have a quantified version that is constant domain and a quantified version that is varying domain. For each of these we have versions with and without equality. And we have predicate abstract extensions for which terms always designate and extensions for which terms might not designate. Rather than give a multiplicity of soundness and completeness arguments in full detail we just summarize what needs to be added to earlier proofs and we do this for a single representative example.

In this chapter we sketch soundness and completeness of the VN tableau system for the modal logic $\mathbf{K}$.

You are invited to pick a different logic beside VN for $\mathbf{K}$, and work through the details for it; this would be a good test of your understanding. In the other direction, you may wish to skip this chapter altogether, on a first pass. It is technical, but it establishes the certainty that our tableau system and our semantics agree. For what follows, it is enough if you believe this. In brief, applications depend on the outcome of what is done here and not on its proof, except for the certainty of that outcome.

16.1 Object Terms Semantically

When we presented tableaus for quantified modal logics in Chap. 8, our formal language needed extending. *Parameters* were introduced, though they only appeared in proofs. They were free variables that could not be quantified—a syntactic condition. How to understand parameters semantically was simple since they were, after all, just a special kind of free variable. For predicate abstracts our language of tableau proofs is once again expanded, but this time the enlargement is not as simple as the addition of parameters. Intensional constant and function symbols can

appear in the formulas we are proving, as parts of predicate abstracts, but we also will see superscripted versions that are only allowed to appear in proofs, and these superscripted versions can be compounded into *Object Terms*, Definition 15.2.1. The semantic behavior for object terms has been discussed a bit, but this was casual and informal. It is now time to move from the informal to the rigorous.

We think of $\sigma \Phi$ as saying that formula Φ is true at possible world σ , where the prefix informally plays the role of a possible world. A non-rigid constant symbol c can appear in a formula we are trying to prove, but in proofs we can also see c^σ . We have been thinking of this informally as the object that c designates at possible world σ . A non-rigid constant symbol c can designate different objects at different worlds, but c^σ represents a particular object, and hence is rigid. It does not matter where you are when you talk about what c designates in world σ . The behavior of c^σ is independent of where an observer might be located, and depends only on the world σ and the constant symbol c . Similar remarks apply to function symbols too. Of course things are actually more complex yet because constant and function symbols might not designate at some worlds.

Thinking of a prefix as a possible world has been rather informal. To make things mathematically precise we will make use of *prefix functions*, a notion which first appeared in Definition 7.4.1 and was extended in Definition 9.3.1. Now we add to the definition of prefix function conditions having to do with non-rigid constant and function symbols. It is this machinery that formalizes our informal ideas. The definitions we are about to give talk about varying domain **K** models, in which terms might not designate. We have presented three other tableau versions of predicate abstraction rules. We discuss soundness and completeness only for the VN version, which happens to be the most complicated. We leave the other three as projects that certainly will test your understanding, but for that reason are worthwhile projects.

Definition 16.1.1 (Prefix Function, Extended Again) Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a varying domain **K** model in which terms might not designate, VN, and let S be a set of prefixed formulas in which object terms are allowed to appear, Definition 15.2.1. We say σ *occurs in* S if it appears in S as a formula prefix, or as a constant or function symbol superscript, or as a parameter subscript. A mapping θ is a *prefix function for* S , in $\mathcal{M}$, using valuation v , if it assigns to each σ that occurs in S some possible world $\theta(\sigma)$ in $\mathcal{G}$, meeting the following conditions.

1. If σ and $\sigma.n$ both occur in S then $\theta(\sigma.n)$ is a world that is accessible from $\theta(\sigma)$, that is, $\theta(\sigma)\mathcal{R}\theta(\sigma.n)$.
2. If the parameter p_σ occurs in a formula of S , $v(p_\sigma) \in \mathcal{D}(\theta(\sigma))$.
3. If $\sigma \langle \lambda x. \Phi \rangle (c)$ occurs in S , where c is a constant symbol then $\mathcal{I}(c, \theta(\sigma))$ is defined, and we say c *designates* at σ .
4. If $\sigma \Phi f(t_1, \dots, t_n)$ occurs in S , where each t_i is a pseudo-closed object term, then $\mathcal{I}(f, \theta(\sigma))$ is defined, and we say f *designates* at σ .

Object terms can contain constant and function symbols, so any semantic understanding of them must involve an interpretation function $\mathcal{I}$ of a model. Object terms can also contain free variables, including parameters, so a valuation function

v for free variables must come in. Finally object terms can contain prefixes, so a prefix function must also come in. We lump all this together into what we call a *context*.

Definition 16.1.2 (Context) A *context* for the evaluation of object terms is a specification of a $\langle \mathbf{K}, \text{with the VN conditions} \rangle$ model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$, a valuation v in $\mathcal{M}$, and a prefix function θ in $\mathcal{M}$.

We now specify how object terms are to be evaluated, and introduce notation for the result of such evaluation. The results do not depend on a choice of possible world, since object terms are meant to be rigid, but they are specified in a context.

Definition 16.1.3 (Object Term Evaluation) Let S be a set of prefixed formulas, where the formulas can contain object terms. Assume we have a context for the evaluation of object terms, where that context involves the model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$, the valuation v in $\mathcal{M}$, and the prefix function θ for S , in $\mathcal{M}$. With respect to this context a mapping is defined, either assigning a value $\llbracket t \rrbracket$ in the domain of the model $\mathcal{M}$ to an object term t , or being undefined on t . The evaluation function is specified as follows.

Category	Definition
Variable	$\llbracket x \rrbracket = v(x)$
Parameter	$\llbracket p_\sigma \rrbracket = v(p_\sigma)$
Constant Symbol	$\llbracket c^\sigma \rrbracket = \begin{cases} \mathcal{I}(c, \theta(\sigma)) & \text{if } c \text{ designates at } \sigma \\ \text{undefined} & \text{otherwise} \end{cases}$
Function Symbol	$\llbracket f^\sigma(t_1, \dots, t_n) \rrbracket = \begin{cases} \mathcal{I}(f, \theta(\sigma))(\llbracket t_1 \rrbracket, \dots, \llbracket t_n \rrbracket) & \text{if } f \text{ designates at } \sigma, \\ & \text{each } \llbracket t_i \rrbracket \text{ is defined,} \\ & \text{and } \langle \llbracket t_1 \rrbracket, \dots, \llbracket t_n \rrbracket \rangle \text{ is in} \\ & \text{the domain of } \mathcal{I}(f, \theta(\sigma)) \\ \text{undefined} & \text{otherwise} \end{cases}$

We say an object term t *designates* in a context provided $\llbracket t \rrbracket$ is defined, and otherwise it does not designate.

Next we say when a formula *containing object terms* is true in a varying domain model, a model in which terms might not designate. A truth definition is for *every possible world*, and not just for those in the range of the prefix function θ . All the parts of Definition 14.6.2 (which in turn builds on earlier definitions) remain the same except for item 1 covering atomic formulas, and item 11 covering function symbols. We begin with the replacement for item 1, provide an example illustrating it, and then give the replacement for item 11.

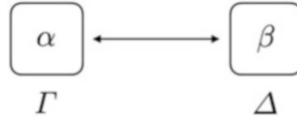
Definition 16.1.4 (Truth in a Non-Rigid Model, in a Context) The following replaces item 1 of Definition 14.6.2, which traces back to Definition 8.6.7. (An appropriate context is assumed.)

1. For an n place relation symbol R ,

- a. $\mathcal{M}, \Gamma \Vdash_v R(x_1, \dots, x_n)$ provided $\langle v(x_1), \dots, v(x_n) \rangle \in \mathcal{I}(R, \Gamma)$. Otherwise $\mathcal{M}, \Gamma \nVdash_v R(x_1, \dots, x_n)$.
- b. If $t_1, \dots, t_n$ are general object terms, $\mathcal{M}, \Gamma \Vdash_v R(t_1, \dots, t_n)$ provided each of $t_1, \dots, t_n$ designates, and $\mathcal{M}, \Gamma \Vdash_w R(x_1, \dots, x_n)$, where $x_1, \dots, x_n$ are previously unused variables, and w is the variant of v such that $w(x_1) = \llbracket t_1 \rrbracket, \dots, w(x_n) = \llbracket t_n \rrbracket$. Otherwise $\mathcal{M}, \Gamma \nVdash_v R(t_1, \dots, t_n)$.

Note that in the definition above, case *1b* reduces things to *1a*. We have expressed it as we did to clearly relate the new condition 1 to the old one in Definitions 14.6.2 and 8.6.7. In fact, both parts could be combined into the following single condition: For object terms, $\mathcal{M}, \Gamma \Vdash_v R(t_1, \dots, t_n)$ provided each of $t_1, \dots, t_n$ designates and $\langle \llbracket t_1 \rrbracket, \dots, \llbracket t_n \rrbracket \rangle \in \mathcal{I}(R, \Gamma)$.

Example 16.1.5 This example illustrates how the definition above works in practice. The diagram shows a two-world modal model $\mathcal{M}$ with each world accessible from the other (though this plays no actual role here). The domain of the world Γ is $\{\alpha\}$, and of the world Δ is $\{\beta\}$.



We only consider a single two-place relation symbol, R , a constant symbol c , and a one-place function symbol f . For these, the interpretation function is as follows.

$$\begin{array}{lll}
 \mathcal{I}(f, \Gamma)(\alpha) & \text{undefined} & \\
 \mathcal{I}(R, \Gamma) = \{\langle \beta, \beta \rangle\} & \mathcal{I}(c, \Gamma) = \alpha & \mathcal{I}(f, \Gamma)(\beta) = \alpha \\
 \mathcal{I}(R, \Delta) = \{\langle \alpha, \beta \rangle\} & \mathcal{I}(c, \Delta) \text{ undefined} & \mathcal{I}(f, \Delta)(\alpha) = \beta \\
 & & \mathcal{I}(f, \Delta)(\beta) = \beta
 \end{array}$$

Assume we have two prefixes, 1 and 1.1, and a prefix function such that $\theta(1) = \Gamma$ and $\theta(1.1) = \Delta$. We first evaluate some representative pseudo-closed object terms, in the context as described.

$$\begin{aligned}
 \llbracket c^1 \rrbracket &= \mathcal{I}(c, \theta(1)) = \mathcal{I}(c, \Gamma) = \alpha \\
 \llbracket f^1(c^1) \rrbracket &= \mathcal{I}(f, \theta(1))(\llbracket c^1 \rrbracket) = \mathcal{I}(f, \Gamma)(\alpha) = \text{undefined} \\
 \llbracket f^{1.1}(c^1) \rrbracket &= \mathcal{I}(f, \theta(1.1))(\llbracket c^1 \rrbracket) = \mathcal{I}(f, \Delta)(\alpha) = \beta
 \end{aligned}$$

$$\llbracket c^{1.1} \rrbracket = \mathcal{I}(c, \theta(1.1)) = \mathcal{I}(c, \Delta) = \text{undefined}$$

$$\llbracket f^1(c^{1.1}) \rrbracket = \mathcal{I}(f, \theta(1))(\llbracket c^{1.1} \rrbracket) = \mathcal{I}(f, \Gamma)(\text{undefined}) = \text{undefined}$$

Now, $\mathcal{M}, \Gamma \Vdash_v R(f^{1.1}(c^1), c^1)$ if $\mathcal{M}, \Gamma \Vdash_w R(x, y)$ where $w(x) = \llbracket f^{1.1}(c^1) \rrbracket = \beta$ and $w(y) = \llbracket c^1 \rrbracket = \beta$, and this is the case. Also a similar calculation shows that $\mathcal{M}, \Delta \Vdash_v R(f^{1.1}(c^1), c^1)$ is not the case, though all of the object terms involved designate. On the other hand, both $\mathcal{M}, \Gamma \Vdash_v R(f^1(c^{1.1}), c^1)$ and $\mathcal{M}, \Delta \Vdash_v R(f^1(c^{1.1}), c^1)$ fail because $f^1(c^{1.1})$ does not designate.

Once the semantic behavior of atomic formulas on object terms has been specified, more complex formula behavior reduces to the atomic level using the usual evaluation rules for propositional connectives, modal operators, and quantifiers, *except* that we do need to supplement the machinery for predicate abstraction involving function symbols. (The machinery for non-rigid constant symbols remains the same.)

Definition 16.1.6 (Truth in a Non-Rigid Model, in a Context) The following replaces item 11 of Definition 14.6.2.

11. For an n -ary intensional function symbol f ,

- a. If f designates at Γ in $\mathcal{G}$ and $\langle v(x_1), \dots, v(x_n) \rangle$ is in the domain of $\mathcal{I}(f, \Gamma)$ then

$$\mathcal{M}, \Gamma \Vdash_v \langle \lambda y. \Phi \rangle (f(x_1, \dots, x_n)) \iff \mathcal{M}, \Gamma \Vdash_w \Phi,$$

where w is the y -variant of v such that $w(y) = \mathcal{I}(f, \Gamma)(v(x_1), \dots, v(x_n))$. Otherwise $\mathcal{M}, \Gamma \nVdash_v \langle \lambda y. \Phi \rangle (f(x_1, \dots, x_n))$.

- b. If f designates at Γ in $\mathcal{G}$ and $t_1, \dots, t_n$ are general pseudo-closed object terms each of which designates then

$$\mathcal{M}, \Gamma \Vdash_v \langle \lambda y. \Phi \rangle (f(t_1, \dots, t_n)) \iff \mathcal{M}, \Gamma \Vdash_w \langle \lambda y. \Phi \rangle (f(x_1, \dots, x_n)),$$

where $x_1, \dots, x_n$ are previously unused variables, and w is the variant of v such that $w(x_1) = \llbracket t_1 \rrbracket, \dots, w(x_n) = \llbracket t_n \rrbracket$. Otherwise $\mathcal{M}, \Gamma \nVdash_v \langle \lambda y. \Phi \rangle (f(t_1, \dots, t_n))$.

Exercises

Exercise 16.1.1 Using the model from Example 16.1.5, evaluate $\llbracket f^{1.1}(c^{1.1}) \rrbracket$.

16.2 A Technical Result

We are leading up to a proof of tableau soundness in the next section, but before we get to it there is a technical but intuitively plausible result that we will need a couple of times. We get this out of the way first. Informally it says that if we replace a pseudo-closed object term with a new variable whose value is the same as that of the object term, it does not affect the overall semantic behavior of whatever it was replaced in. The actual details of the proof are a bit fussy, and we leave much of the work to you. If the content of the proposition is sufficiently clear, no harm will be done if you skip the proof on a first (or second) pass.

Proposition 16.2.1 *Assume we have a context for object term evaluation, consisting of a varying domain model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ in which terms might not designate, a prefix function θ , and a valuation v . Let t_0 be a particular pseudo-closed object term that designates in this context, and so has $\llbracket t_0 \rrbracket$ as its value. Let $v[x/\llbracket t_0 \rrbracket]$ be the valuation that is the x -variant of v whose value at x is $\llbracket t_0 \rrbracket$.*

Having introduced the valuation $v[x/\llbracket t_0 \rrbracket]$, we now have a second evaluation context, just like the first except that we use valuation $v[x/\llbracket t_0 \rrbracket]$ instead of v . In what follows, we write the value of an arbitrary object term u that is computed using this second context as $\llbracket u \rrbracket$. We have the following connections between the two contexts.

1. *If $s(x)$ is an object term with a free variable x , and $s(t_0)$ is the result of replacing all free occurrences of x in $s(x)$ with occurrences of t_0 , then $\llbracket s(t_0) \rrbracket = \llbracket s(x) \rrbracket$.*
2. *If $\varphi(x)$ is a formula with a free variable x , and $\varphi(t_0)$ is the result of replacing all free occurrences of x in $\varphi(x)$ with occurrences of t_0 , then $\mathcal{M}, \theta(\sigma) \Vdash_v \varphi(t_0) \iff \mathcal{M}, \theta(\sigma) \Vdash_{v[x/\llbracket t_0 \rrbracket]} \varphi(x)$.*

Proof For part 1, a proper proof is by complete induction on the complexity of $s(x)$. Rather than presenting this in full, we look at what should be a sufficiently illustrative example. Suppose σ and τ are prefixes, f is a one-place function symbol, and $s(x)$ is $f^\sigma(f^\tau(x))$. Then we have the following.

$$\begin{aligned}
 \llbracket s(t_0) \rrbracket &= \llbracket f^\sigma(f^\tau(t_0)) \rrbracket \\
 &= \mathcal{I}(f, \theta(\sigma))(\llbracket f^\tau(t_0) \rrbracket) && \text{Definition 16.1.3} \\
 &= \mathcal{I}(f, \theta(\sigma))(\mathcal{I}(f, \theta(\tau))(\llbracket t_0 \rrbracket)) && \text{Definition 16.1.3} \\
 &= \mathcal{I}(f, \theta(\sigma))(\mathcal{I}(f, \theta(\tau))(\llbracket x \rrbracket)) && \text{Definition of } v(x/\llbracket t \rrbracket) \\
 &= \mathcal{I}(f, \theta(\sigma))(\llbracket f^\tau(x) \rrbracket) && \text{Definition 16.1.3} \\
 &= \llbracket f^\sigma(f^\tau(x)) \rrbracket && \text{Definition 16.1.3} \\
 &= \llbracket s(x) \rrbracket
 \end{aligned}$$

For part 2, the proof is by complete induction on the complexity of the formula $\varphi(x)$. There are multiple cases and we just look at just one, predicate abstraction

involving a function symbol. To keep the notation simple, we take the function symbol to be one-place. Let $\varphi(x)$ be the formula $\langle \lambda y. \Phi(x, y) \rangle (f(-x-))$, where $-x-$ is an (oddly written) object term having free occurrences of the variable x , and $\Phi(x, y)$ likewise contains free occurrences of x and y . If f does not designate at $\theta(\sigma)$ the result is immediate since both sides of the equivalence in part 2 will be false. Now assume f designates at $\theta(\sigma)$ and assume, as induction hypothesis, that the result is known for formulas simpler than $\varphi(x)$.

Our proof works from both ends to the middle, so to speak. That is, we simplify both $\mathcal{M}, \theta(\sigma) \Vdash_v \varphi(t_0)$ and $\mathcal{M}, \theta(\sigma) \Vdash_{v[x/\llbracket t_0 \rrbracket]} \varphi(x)$, through sequences of equivalent statements, winding up with the same thing in both cases, thus establishing the desired result. We start from the left.

$$\mathcal{M}, \theta(\sigma) \Vdash_v \varphi(t_0) \iff \mathcal{M}, \theta(\sigma) \Vdash_v \langle \lambda y. \Phi(t_0, y) \rangle (f(-t_0-)) \quad (16.1)$$

$$\iff \mathcal{M}, \theta(\sigma) \Vdash_{v'} \Phi(t_0, y) \quad (16.2)$$

where v' is the y variant of v such that $v'(y) = \mathcal{I}(f, \sigma)(\llbracket -t_0 - \rrbracket)$. This is by Definition 16.1.6, using Definition 16.1.3 to evaluate $f(-t_0-)$. By the induction hypothesis we have the following, continuing from (16.2).

$$\mathcal{M}, \theta(\sigma) \Vdash_{v'} \Phi(t_0, y) \iff \mathcal{M}, \theta(\sigma) \Vdash_{v'[x/\llbracket t_0 \rrbracket]} \Phi(x, y) \quad (16.3)$$

Now suppose we work backwards from the right end. Thus we begin with the following.

$$\mathcal{M}, \theta(\sigma) \Vdash_{v[x/\llbracket t_0 \rrbracket]} \varphi(x) \iff \mathcal{M}, \theta(\sigma) \Vdash_{v[x/\llbracket t_0 \rrbracket]} \langle \lambda y. \Phi(x, y) \rangle (f(-x-)) \quad (16.4)$$

Once again we apply Definition 16.1.6.

$$\mathcal{M}, \theta(\sigma) \Vdash_{v[x/\llbracket t_0 \rrbracket]} \langle \lambda y. \Phi(x, y) \rangle (f(-x-)) \iff \mathcal{M}, \theta(\sigma) \Vdash_w \Phi(x, y), \quad (16.5)$$

where w is the y variant of $v[x/\llbracket t_0 \rrbracket]$ with $w(y)$ being the result of evaluating $f(-x-)$ using the valuation $v[x/\llbracket t_0 \rrbracket]$ itself. Using the notation introduced in the statement of the Proposition, $w(y) = \mathcal{I}(f, \sigma)(\llbracket -x - \rrbracket)$.

Now we show that the valuations w and $v'[x/\llbracket t_0 \rrbracket]$ are actually the same things. There are two cases.

First, let z be a variable that is not y . We have $w(z) = v[x/\llbracket t_0 \rrbracket](z)$, since z is not y . And $v[x/\llbracket t_0 \rrbracket](z) = v'[x/\llbracket t_0 \rrbracket](z)$ because v and v' are y variants and z is not y .

Second, we consider y itself. We have $w(y) = \mathcal{I}(f, \sigma)(\llbracket -x - \rrbracket)$. By part 1 this is $\mathcal{I}(f, \sigma)(\llbracket -t_0 - \rrbracket)$, which is $v'(y)$. But $v'(y) = v'[x/\llbracket t_0 \rrbracket](y)$, since v' and $v'[x/\llbracket t_0 \rrbracket]$ can only differ on x .

Thus w and $v'[x/\llbracket t_0 \rrbracket]$ are the same valuation, and so we have the following.

$$\mathcal{M}, \theta(\sigma) \Vdash_w \Phi(x, y) \iff \mathcal{M}, \theta(\sigma) \Vdash_{v'[x/\llbracket t_0 \rrbracket]} \Phi(x, y) \quad (16.6)$$

Now, combining (16.1), (16.2), (16.3), and (16.6), (16.5), (16.4), we have the promised equivalence. Other cases of the induction are left to the reader.

16.3 Soundness for $\mathbf{K}$ (VN Version)

We are at the point where we can prove soundness for the VN version of $\mathbf{K}$. In fact we prove something stronger, namely that we have soundness even with the Substitution Rule for Equality extended to allow the formula being substituted in to be non-atomic. This actually is what we must do, in order to show that general, and not just atomic, substitutivity is an admissible rule. The proof of Proposition 11.5.4 should be read at this point.

There are a few simple facts it will help to have on hand, and we get these out of the way first.

Proposition 16.3.1 *Every prefixed formula appearing in a tableau starting with $1 \neg\Phi$, where Φ is a closed formula, is a pseudo-closed formula.*

Proof This is true of the initial prefixed signed formula starting a tableau proof because it involves a closed formula. Then one simply notes that if the premises of any of the tableau rules involve only pseudo-closed formulas, the same is true of the conclusion.

Thus the presence of free variables other than parameters in a proof need not concern us.

Tableau soundness proofs begin by saying what it means for a tableau to be satisfiable. We have more syntactical machinery now than we had in earlier chapters, and so the definition of satisfiability is more complicated than it was. And we also modified our notion of prefix function, and we have taken object terms into account.

The following definition of satisfiability is specifically for varying domain $\mathbf{K}$, with the VN assumptions.

Definition 16.3.2 (K Tableau Satisfiability, VN Conditions) Let $\mathcal{T}$ be a $\mathbf{K}$ tableau meeting the VN conditions. As usual, $\mathcal{T}$ is satisfiable if some branch $\mathcal{B}$ is, and this is the case if the set of prefixed formulas S on $\mathcal{B}$ is satisfiable. It is this last part that we now define.

A set S of prefixed pseudo-closed formulas is satisfiable if there is a context (Definition 16.1.2) consisting of a $\mathbf{K}$ model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ meeting the VN conditions, a valuation v in $\mathcal{M}$, and a prefix function θ in $\mathcal{M}$, that assigns a value to those prefixes that appear in formulas of S , and for each $\sigma \Phi \in S$ we have $\mathcal{M}, \theta(\sigma) \Vdash_v \Phi$.

Let Φ be a closed formula allowing predicate abstracts but not pseudo-closed object terms, which are not part of our basic modal language. Soundness is proved in contrapositive form: if Φ is not valid (in $\mathbf{K}$, varying domain, where terms might

not designate), then Φ does not have a tableau proof. Tableau soundness proofs all follow the same strategy, which we briefly recall.

Assuming Φ is not valid, there is some model $\mathcal{M}$ and some possible world Γ in it such that $\mathcal{M}, \Gamma \not\models_v \Phi$ (the details of the valuation v don't matter, since Φ is closed). Then the set $\{1 \neg \Phi\}$ is satisfiable, using the prefix function θ mapping 1 to Γ . Next one shows that satisfiability is preserved by every tableau rule application. Since a closed tableau cannot be satisfiable, one can never produce a tableau proof starting with $1 \neg \Phi$, so Φ is not provable.

The heart of the soundness proof just sketched is the verification that if a tableau rule is applied to a satisfiable tableau, the result is another satisfiable tableau. Once this is established, the soundness argument is finished. As usual, it is established by giving arguments that each of the tableau rules preserves satisfiability, with different arguments for different rules. For tableau rules that do not involve equality or predicate abstraction this is exactly as it was in Proposition 9.3.3, and is not repeated here. What needs proving now is that preservation of satisfiability applies to the predicate abstraction rules and to the rules for equality. This gives us several new cases, and for each case the assumptions are the same.

General Assumption Branch $\mathcal{B}$ of tableau $\mathcal{T}$ is satisfiable in the context consisting of model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$, prefix function θ , and valuation v .

Now, here are the cases that we have not covered earlier.

16.3.1 Predicate Abstraction

We discuss the two rules from Definition 15.5.9 involving function symbols. The rules for constant symbols are similar but easier. In what follows, f is an n -place function symbol and $t_1, \dots, t_n$ are pseudo-closed object terms.

$$\frac{\sigma \langle \lambda x. \Phi(x) \rangle (f(t_1, \dots, t_n))}{\sigma \Phi(f^\sigma(t_1, \dots, t_n))} \qquad \frac{\sigma \neg \langle \lambda x. \Phi(x) \rangle (f(t_1, \dots, t_n))}{\sigma \neg \Phi(f^\sigma(t_1, \dots, t_n))}$$

provided $f^\sigma(t_1, \dots, t_n)$ is positively
generated on the tableau branch

16.3.1.1 The Positive Abstract Rule

To keep notational clutter down we take n to be 1, which is enough to illustrate the essential ideas. Thus, we show the following rule preserves satisfiability.

$$\frac{\sigma \langle \lambda x. \Phi(x) \rangle (f(t))}{\sigma \Phi(f^\sigma(t))}$$

We assume $\sigma \langle \lambda x. \Phi(x) \rangle (f(t))$ occurs on the satisfiable tableau branch $\mathcal{B}$, and we show the result of adding $\sigma \Phi(f^\sigma(t))$ to $\mathcal{B}$ is again a satisfiable branch. Here t is a pseudo-closed object term.

It is important to point out that the notation used in the rule above is very much overloaded. In the formula above the line the occurrence of “ $(f(t))$ ” is an explicit part of the expression. There is a left parenthesis followed by f followed by a left parenthesis followed by the symbols making up the term t followed by two right parentheses. But in the formula below the line the occurrence of “ $(f^\sigma(t))$ ” does not play the same role. Instead it indicates that below the line we have taken the prefixed formula $\sigma \Phi(x)$ and replaced all free occurrences of x in Φ with occurrences of the string of symbols making up $f^\sigma(t)$. Unfortunately it is easy to confuse these two usages.

Using our General Assumption that branch $\mathcal{B}$ is satisfiable, and the assumption that $\sigma \langle \lambda x. \Phi(x) \rangle (f(t))$ is on branch $\mathcal{B}$, we have the following.

$$\mathcal{M}, \theta(\sigma) \Vdash_v \langle \lambda x. \Phi(x) \rangle (f(t)) \quad (16.7)$$

Applying the various parts of Definition 16.1.6 to (16.7), we conclude that t must designate, f must designate at $\theta(\sigma)$ and,

$$\mathcal{M}, \theta(\sigma) \Vdash_{v[x/\llbracket f^\sigma(t) \rrbracket]} \Phi(x), \quad (16.8)$$

where $v[x/\llbracket f^\sigma(t) \rrbracket]$ is the valuation that is like v except that it maps x to $\llbracket f^\sigma(t) \rrbracket$. (We are making use of notation from the statement of Proposition 16.2.1.) Now, applying part 2 of Proposition 16.2.1, we have

$$\mathcal{M}, \theta(\sigma) \Vdash_v \Phi(f^\sigma(t)) \quad (16.9)$$

and this establishes that the result of adding $\sigma \Phi(f^\sigma(t))$ to branch $\mathcal{B}$ is again a satisfiable branch.

16.3.1.2 The Negative Abstract Rule

Again we take n to be 1 to keep notation simple. We show the following rule preserves satisfiability.

$$\frac{\sigma \neg \langle \lambda x. \Phi(x) \rangle (f(t))}{\sigma \neg \Phi(f^\sigma(t))} \quad \text{provided } f^\sigma(t) \text{ is positively generated on the tableau branch}$$

One assumption is that $\sigma \neg \langle \lambda x. \Phi(x) \rangle (f(t))$ occurs on the satisfiable branch $\mathcal{B}$ so by our General Assumption, we have the following.

$$\mathcal{M}, \theta(\sigma) \Vdash_v \neg \langle \lambda x. \Phi(x) \rangle (f(t)) \quad (16.10)$$

or equivalently,

$$\mathcal{M}, \theta(\sigma) \nVdash_v \langle \lambda x. \Phi(x) \rangle (f(t)). \quad (16.11)$$

A second assumption is that $f^\sigma(t)$ is positively generated somewhere on branch $\mathcal{B}$. Since the branch is satisfiable, it follows by the argument for the Positive Abstract Rule that $f^\sigma(t)$ designates. Then in order to have (16.11), it must be that

$$\mathcal{M}, \theta(\sigma) \nVdash_v \Phi(f^\sigma(t)) \quad (16.12)$$

and thus

$$\mathcal{M}, \theta(\sigma) \Vdash_v \neg \Phi(f^\sigma(t)). \quad (16.13)$$

This establishes that the extension of $\mathcal{B}$ with $\sigma \neg \Phi(f^\sigma(t))$ is satisfiable.

16.3.1.3 Equality, The Reflexivity Rule

The Reflexivity Rule is in Definition 15.5.10. Recall, model $\mathcal{M}$ is normal, and branch $\mathcal{B}$ of a tableau is satisfiable in $\mathcal{M}$, using valuation v and prefix function θ . We show that adding $\sigma (t = t)$ to the branch preserves satisfiability, provided: σ is already on the branch, and t is a pseudo-closed object term that is a parameter or is positively generated on the branch $\mathcal{B}$.

The conditions that t must satisfy are enough to ensure that t designates, and $\llbracket t \rrbracket$ is defined. Since σ is on branch $\mathcal{B}$, then $\theta(\sigma)$ is also defined. Then since $\mathcal{M}$ is normal, $\mathcal{M}, \theta(\sigma) \Vdash_v (t = t)$ is the case because $\llbracket t \rrbracket$ is self-identical. Thus the extended branch is satisfiable.

16.3.1.4 Equality, The General Substitutivity Rule

The Atomic Substitutivity Rule is given in Definition 15.5.3. There is also a *General* version which we now state, and show it preserves satisfiability (and hence so does the Atomic version). Informally, this tells us the General version is safe to use in tableau proofs. When we show completeness, it will be under the condition that Substitutivity is atomic, and this tells us the General version is not actually

needed. Here is the General Substitutivity version. For an *arbitrary* formula $\Phi(x)$, not necessarily atomic, the following tableau branch extension rule is allowed.

$$\frac{\tau(t = u) \quad \sigma \Phi(t)}{\sigma \Phi(u)}$$

Our assumptions are that branch $\mathcal{B}$ of a tableau is satisfiable in model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$, using valuation v and prefix function θ . And also, both $\tau(t = u)$ and $\sigma \Phi(t)$ are on the branch, where t and u are pseudo-closed object terms. We show that adding $\sigma \Phi(u)$ to branch $\mathcal{B}$ preserves satisfiability. Our main tool is Proposition 16.2.1, which should be consulted for notation.

Because $\tau(t = u)$ is on branch $\mathcal{B}$, which is satisfiable, we have $\llbracket t \rrbracket = \llbracket u \rrbracket$. (This uses the fact that $\mathcal{M}$ is normal and the equality symbol is interpreted by actual equality.) Likewise since $\sigma \Phi(t)$ is on $\mathcal{B}$, we have $\mathcal{M}, \theta(\sigma) \Vdash_v \Phi(t)$. Then by Proposition 16.2.1 part 2, we have $\mathcal{M}, \theta(\sigma) \Vdash_{v[x/\llbracket t \rrbracket]} \Phi(x)$. Since $\llbracket t \rrbracket = \llbracket u \rrbracket$, this gives us $\mathcal{M}, \theta(\sigma) \Vdash_{v[x/\llbracket u \rrbracket]} \Phi(x)$, and by Proposition 16.2.1 part 2 again, we have $\mathcal{M}, \theta(\sigma) \Vdash_v \Phi(u)$, which says that the extended branch is satisfiable.

Exercises

Exercise 16.3.1 We have proved the soundness of the **K** rules meeting the *VN* condition. Your problem is to prove soundness for one or more of the rest, *VA*, *CA*, and *CN*. Of these, *VA* is a simple variation of what we did, but the *CA* and *CN* rules require more thought.

16.4 Hintikka Sets for **K** (VN Version)

As the title says, this is for **K**, varying domain, where terms might not designate. Other logics and conditions require modifications which we leave to the reader. Over the course of this book we have introduced a succession of Hintikka set definitions, each building on previous versions. Propositional and modal conditions were in Definition 7.5.1, quantifier conditions were in Definition 9.4.1, and equality conditions were in Definition 11.7.1. The following combines all these, with some obvious modifications. The underlying language now is understood to be first-order with equality and with the inclusion of constant and function symbols and predicate abstraction machinery, as well as object terms. And the equality conditions have been modified so that their application is extended to terms beyond just parameters. The *VN* rules from Sect. 15.5 are the relevant ones throughout. The numbering of the clauses is the same as it has been throughout the preceding chapters.

Definition 16.4.1 (K Hintikka Set, VN Conditions) We transfer to Hintikka sets some terminology from Definition 15.3.1. For a non-rigid constant symbol c , c^σ is positively generated in a Hintikka set H if a prefixed formula $\sigma \langle \lambda x. \Phi \rangle (c)$ is present in H for some Φ . Likewise for an n -place function symbol f , $f^\sigma(t_1, \dots, t_n)$ is positively generated in H if $\sigma \langle \lambda x. \Phi \rangle (f(t_1, \dots, t_n))$ is present for some Φ .

A set H of prefixed pseudo-closed formulas is a *Hintikka set* (in the **K**, VN sense) if it meets the following conditions.

H-1 Not both σA and $\sigma \neg A$ are present, for any atomic formula A .

H-2 The following propositional closure conditions are met.

$$\begin{aligned} \sigma \Phi \wedge \Psi \in H &\implies \sigma \Phi \in H \text{ and } \sigma \Psi \in H \\ \sigma \neg(\Phi \wedge \Psi) \in H &\implies \sigma \neg\Phi \in H \text{ or } \sigma \neg\Psi \in H \\ \sigma \Phi \vee \Psi \in H &\implies \sigma \Phi \in H \text{ or } \sigma \Psi \in H \\ \sigma \neg(\Phi \vee \Psi) \in H &\implies \sigma \neg\Phi \in H \text{ and } \sigma \neg\Psi \in H \\ \sigma \Phi \supset \Psi \in H &\implies \sigma \neg\Phi \in H \text{ or } \sigma \Psi \in H \\ \sigma \neg(\Phi \supset \Psi) \in H &\implies \sigma \Phi \in H \text{ and } \sigma \neg\Psi \in H \\ \sigma \neg\neg\Phi \in H &\implies \sigma \Phi \in H \end{aligned}$$

H-3 The following modal closure conditions are met.

$$\begin{aligned} \sigma \Box\Phi \in H &\implies \sigma.n \Phi \in H \text{ for every } \sigma.n \text{ appearing in } H \\ \sigma \neg\Diamond\Phi \in H &\implies \sigma.n \neg\Phi \in H \text{ for every } \sigma.n \text{ appearing in } H \\ \sigma \Diamond\Phi \in H &\implies \sigma.n \Phi \in H \text{ for some } \sigma.n \\ \sigma \neg\Box\Phi \in H &\implies \sigma.n \neg\Phi \in H \text{ for some } \sigma.n \end{aligned}$$

H-4 There are no special modal closure conditions since we are only considering **K**.

H-5 The following quantifier closure conditions are met.

$$\begin{aligned} \sigma (\exists x)\Psi(x) \in H &\implies \sigma \Psi(p_\sigma) \in H \text{ for some parameter } p_\sigma \\ \sigma \neg(\forall x)\Psi(x) \in H &\implies \sigma \neg\Psi(p_\sigma) \in H \text{ for some parameter } p_\sigma \\ \sigma (\forall x)\Psi(x) \in H &\implies \sigma \Psi(p_\sigma) \in H \text{ for every parameter } p_\sigma \\ \sigma \neg(\exists x)\Psi(x) \in H &\implies \sigma \neg\Psi(p_\sigma) \in H \text{ for every parameter } p_\sigma \end{aligned}$$

H-6 The following two equality conditions are met.

- a. If σ is a prefix that occurs in H , and t is a parameter or a pseudo-closed object term that is positively generated in H , then $\sigma (t = t) \in H$.

- b. For any pseudo-closed object terms t and u , and for any *atomic* formula $\Phi(x)$, if $\sigma(t = u) \in H$ and $\tau \Phi(t) \in H$ then $\tau \Phi(u) \in H$.

H-7 The following predicate abstraction conditions are met. For any constant symbol c , n -place function symbol f , and pseudo-closed object terms $t_1, \dots, t_n$:

$$\begin{aligned}
 & \sigma \langle \lambda x. \Phi(x) \rangle (c) \in H \implies \sigma \Phi(c^\sigma) \in H \\
 & \left. \begin{array}{l} \sigma \neg \langle \lambda x. \Phi(x) \rangle (c) \in H \\ \text{and} \\ c^\sigma \text{ is positively generated in } H \end{array} \right\} \implies \sigma \neg \Phi(c^\sigma) \in H \\
 & \sigma \langle \lambda x. \Phi(x) \rangle (f(t_1, \dots, t_n)) \in H \implies \sigma \Phi(f^\sigma(t_1, \dots, t_n)) \in H \\
 & \left. \begin{array}{l} \sigma \neg \langle \lambda x. \Phi(x) \rangle (f(t_1, \dots, t_n)) \in H \\ \text{and} \\ f^\sigma(t_1, \dots, t_n) \text{ is positively generated in } H \end{array} \right\} \implies \sigma \neg \Phi(f^\sigma(t_1, \dots, t_n)) \in H
 \end{aligned}$$

We conclude this section with a few simple remarks. First, in H-6 part *b* there is a restriction on substitutivity, to atomic formulas. This relates to earlier remarks in Sect. 11.5, where the significance of this restriction was discussed. We have already shown that our tableau rules are sound allowing General Substitutivity. Now we are entering on the task of proving they are complete allowing only Atomic Substitutivity, and this why the atomic restriction appears in H-6 part *b*.

The Reflexivity condition, H-6 part *a*, also has an important consequence. Any pseudo-closed object term t that is positively generated in H must actually appear in a prefixed atomic formula in H , specifically in $\sigma(t = t)$. Likewise, since the only pseudo-closed object terms that can appear must be positively generated, we have that the positively generated pseudo-closed object terms are exactly the ones that occur in prefixed atomic formulas.

Finally, it is easy to check that every Hintikka set condition, except for H-1 and H-6, involves a move that reduces degree, Definition 14.1.2.

Exercises

Exercise 16.4.1 Suppose H is a **K** Hintikka set meeting the *VN* conditions, and t , u , and v are pseudo-closed object terms.

1. Show that if $\sigma(t = u) \in H$ and τ is a prefix that occurs in H , then $\tau(t = u) \in H$.
2. Show that if $\sigma(t = u) \in H$ then $\sigma(u = t) \in H$.
3. Show that if $\sigma(t = u)$ and $\sigma(u = v)$ are in H then $\sigma(t = v)$ is in H .

16.5 Adding A Witness World

We want to prove a version of Hintikka's Lemma: every Hintikka set (as currently defined) is satisfiable in an appropriate model. Generally speaking, previous constructions still work well and extend naturally to cover predicate abstracts. However there is a curious problem that comes up for the first time now. Recall that in earlier proofs we created models in which the possible worlds were formula prefixes, and for varying domain semantics the domain of each possible world was the set of parameters having that world as its subscript. Keep this in mind for the following discussion.

Suppose we have a Hintikka set H containing $\sigma \langle \lambda x. \Phi(x) \rangle (c)$. By item (H-7) of Definition 16.4.1, H must also contain $\sigma \Phi(c^\sigma)$. As we just noted, we have previously been using parameters to supply us the objects of the model we are constructing, but what about pseudo-closed object terms—they should designate objects of the model too. In the present case we could simply announce that c^σ is an object in the model, as well as being a term in our formal language, one that designates itself in the same way that parameters do. Then c^σ would literally be what the constant c designates at the possible world σ . In fact, this is what we will do. But in our VN semantics, what is designated at a world need not be an existent object at that world, and this presents a problem for us. c^σ should be an object in the model we are constructing, but it need not be an object in the domain of world σ . Then, where is it?

The problem for model construction just raised is not a rare issue. For instance, if possible worlds are instants of time throughout the lifetime of an author, and c is intended to be understood as *the next book the author writes*, there may be many instants at which c designates, but it can never designate something that exists at that instant. This particular example is rather easy to deal with. Assuming that *the next book the author writes* designates now, it simply designates something that exists later. This is a not uncommon real-world situation. But in our formal tableau setting, if we instantiate a predicate abstract involving constant symbol c during the course of a tableau proof we get an object term, c^σ , and it should designate an object that, according to our semantics, must exist in some possible world. But, which one? In general, an attempted tableau proof will provide us with no useful information about this.

One way that is common in the literature for handling this problem is to allow models to have an inner and an outer model domain, where the inner domain consists of existents and the outer domain consists of meaningful objects that don't happen to exist (or if they do, we don't know where). This is the approach taken by *free logic*, something we discussed a bit in Sect. 8.8. If we were to use this approach here, the inner domain would be the union of the domains of all the possible worlds of our model—the things that exist somewhere—and this would simply be the set of all parameters. But there would also be a larger outer domain of objects that might or might not actually exist anywhere. Formally, this outer domain would include not just parameters but all pseudo-closed object terms. We have decided not to follow

this approach; for us the only objects there are in our models are things that actually exist somewhere—in some possible world. In our semantics, something must be an existent in a possible world to be an object we can ‘talk about’ in any possible world.

We must emphasize that either approach—having an outer domain or having a more extensive set of possible worlds—is a reasonable one. Both work, and both are of about the same complexity of formulation. In developing formal logics, sometimes things are forced on us by the subject matter, and sometimes one has design choices to make. Our choice is, loosely speaking, to assume that the objects our models can ‘talk about’ are the objects that are possible existents, by which we mean that they exist in some possible world.

Our solution to the problem of what to do when we don’t have information about where a designating term designates an existent is a bit of a cheat. We will simply create an artificial possible world whose domain is what would have been the outer domain if we had followed the free logic approach. But we will ensure that this world is inaccessible from all the rest of the possible worlds we have. We call such a world a ‘witness’ world. It is purely a semantic device and we will not change our definitions of tableau system or proof. But we do need to play with the structure of Hintikka sets so that we get what we need for the construction of our kind of models, containing a witness world.

We now describe a simple method for ‘padding out’ a Hintikka set H to a new Hintikka set we call H^w . Prefixes have been with us in this book since Definition 7.1.1. They are just finite sequences of positive integers. You probably have gotten used to the convention that they always start with the integer 1. This is, in fact, a consequence of Definition 7.1.5, that a proof of Φ starts with $1 \neg\Phi$, but there is nothing essential about all this. We now simply allow 2 as a prefix, in addition to what we have been using. We call it our *witness prefix*. Starting with Hintikka set H , we expand it to another Hintikka set H^w that allows formulas prefixed with 2. We refer to the prefixes coming from H as *ordinary prefixes*. Note that for the witness prefix 2 and any ordinary prefix σ beginning with 1, neither $\sigma.n = 2$ nor $2.n = \sigma$ for any n .

16.5.1 Construction of H^w from H

Let H be a varying domain Hintikka set with equality and predicate abstraction, meeting the VN conditions. We assume all prefixes in H begin with 1. The prefixed formulas added to expand H to H^w are just all those of the form $2(t = u)$, where t and u are pseudo-closed object terms such that $\sigma(t = u)$ occurs in H for some ordinary prefix σ (or equivalently, such that $\sigma(t = u)$ occurs in H for *every* ordinary prefix σ , by Exercise 16.4.1, part 1). Consequently $\sigma(t = u)$ is in H^w for some prefix (ordinary or witness) if and only if $\sigma(t = u)$ is in H^w for every prefix (ordinary or witness) σ if and only if $\sigma(t = u)$ is in H for every ordinary prefix σ .

It is very easy to see that when starting with a Hintikka set H , the expansion to H^w will be another Hintikka set. Condition H-1 is trivially met since it is already the case for members of H , and there are no negated formulas in H^w with 2 as a prefix. Also since every prefixed formula added to H to produce H^w is atomic, all the Hintikka set conditions H-2 through H-5 and H-7 trivially still hold since they are conditionals whose antecedents fail on members of H^w that are not in H .

For part *a* of H-6, if t is a pseudo-closed object term that is positively generated in H^w , it must be positively generated in H itself since no new predicate abstracts were added in the extension H^w . But then since H is a Hintikka set, $\sigma(t = t)$ is in H for every ordinary prefix, and so these as well as $2(t = t)$ are in H^w .

For part *b* of H-6, suppose $\sigma(t = u)$ and $\tau\Phi(t)$ are in H^w . From the first of these, $\sigma(t = u)$ is in H for every ordinary prefix σ . Since $\tau\Phi(t) \in H^w$, if τ is ordinary, $\tau\Phi(t) \in H$, so $\tau\Phi(u) \in H$ since H is a Hintikka set, and then $\tau\Phi(t) \in H^w$. If τ is 2, it must be that $\Phi(t)$ is an equality statement, since these were the only things added to H to form H^w . Then $\Phi(t)$ must occur in H with every ordinary prefix. So again since H is a Hintikka set, $\Phi(u)$ will occur in H with every ordinary prefix, and so by definition, $2\Phi(u)$ will be in H^w .

16.6 Hintikka's Lemma, First Pass

Our work now follows the general outlines of Sect. 11.7. Ultimately we want a normal model, where the interpretation of the equality symbol is by the equality relation. If you recall, a direct approach produces a model that is not normal, but this model turns out to have the properties we need to convert it into a normal model. We do the first part of this now, and discuss the conversion into a normal model in the following section.

Proposition 16.6.1 (Hintikka's Lemma, Part 1) *Let H be a first-order $\mathbf{K}$ Hintikka set with equality and predicate abstraction, meeting the VN conditions (in which all prefixes are rooted at 1). The Hintikka set H^w that extends H by adding a witness prefix, as described in Sect. 16.5, is satisfiable in a varying domain $\mathbf{K}$ model in which terms might not designate.*

Proof Let H be a Hintikka set meeting the conditions of Definition 16.4.1, and rooted at 1. We saw in Sect. 16.5 how to extend H to a Hintikka set H^w in which there is a witness prefix, 2. We show satisfiability of H^w . The use of H^w ensures that we get a model in which every non-rigid term that must designate somewhere designates an existent in the witness world. We use a variation of the model construction described in the proof of Proposition 9.4.3, producing a model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ in which H^w is satisfiable, though it is not a normal model.

As usual, $\mathcal{G}$ is the set of prefixes that occur in H^w , but now this includes both ordinary and witness prefixes. We set $\sigma \mathcal{R} \sigma.n$ whenever the prefixes σ and $\sigma.n$ are

both in $\mathcal{G}$. All this is as in previous chapters, except that now there is a witness prefix/possible world, 2, which is not related to any member of $\mathcal{G}$ using $\mathcal{R}$.

We also need a *prefix function*, Definition 16.1.1, but this is simple since our possible worlds are prefixes themselves. We simply require that $\theta(\sigma) = \sigma$ for each prefix. In what follows, to keep notation simple, we generally omit mention of the prefix function, understanding prefixes to have both a syntactic and a semantic role.

For domains of quantification, ordinary prefixes are as they were in the proof of Proposition 9.4.3, but the witness prefix is treated differently. In detail, if σ is an ordinary prefix, $\mathcal{D}(\sigma)$ is the set of all parameters with subscript σ , while for the witness parameter, $\mathcal{D}(2)$ is the set of all parameters together with all pseudo-closed terms t that are positively generated in H^w (or equivalently that are positively generated in H). Note that the members of $\mathcal{D}(2)$ play a dual role, just as parameters do in $\mathcal{D}(\sigma)$ where σ is an ordinary prefix. They are both syntactic objects in our formal language, and they are members of domains of possible worlds in our model $\mathcal{G}$. We note that the domain of the model is simply $\mathcal{D}(2)$ itself.

Finally we specify an interpretation function $\mathcal{I}$, completing the definition of our model.

Atomic Formulas For each n -place relation symbol P , and each prefix σ (ordinary or witness)

$$\mathcal{I}(P, \sigma) = \{\langle t_1, \dots, t_n \rangle \mid \sigma P(t_1, \dots, t_n) \text{ occurs in } H^w\}.$$

Constant Symbols A constant symbol c designates at σ (ordinary or witness) if c^σ is positively generated in H^w (or equivalently, in H), and if it does, $\mathcal{I}(c, \sigma)$ is the object term c^σ .

Function Symbols An n -place function symbol f designates at σ (ordinary or witness) and has $\langle t_1, \dots, t_n \rangle$ in its domain if $f^\sigma(t_1, \dots, t_n)$ is positively generated in H^w (or in H), and if so, the value of $\mathcal{I}(f, \sigma)(t_1, \dots, t_n)$ is the object term $f^\sigma(t_1, \dots, t_n)$.

In Definition 16.1.3 we said how to evaluate object terms in a model, in a *context* consisting of an interpretation function, a prefix function, and a valuation function. For the model $\mathcal{M}$ we just constructed, this gives us the following rather nice item. The proof is simple and is left as an exercise. Recall that being positively generated in a Hintikka set was characterized in Definition 16.4.1.

Fact 16.6.2 *Let θ be the prefix function that maps each prefix to itself. Let v_0 be a valuation that maps every parameter to itself. And let $\mathcal{I}$ be the interpretation specified above. In the resulting context, if t is a pseudo-closed object term that is positively generated in H^w , then t designates in the model, and $\llbracket t \rrbracket = t$.*

We now have the model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$. Here is the fundamental fact about it. Establishing it completes the proof of Proposition 16.6.1.

Fact 16.6.3 *The Hintikka set H^w is satisfiable in the model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$, as defined above.*

What follows is a sketch of the proof of this Fact. Define a valuation v_0 by setting $v_0(p_\sigma) = p_\sigma$ for each parameter p_σ and letting v_0 be arbitrary on free variables that are not parameters. (This doesn't matter since the only free variables that appear in a Hintikka set are parameters.) Also let θ be the usual prefix mapping sending members of $\mathcal{G}$ to themselves. We continue the practice of omitting explicit mention of θ and think of a prefix as having both a syntactic and a semantic role.

Before getting to the details of the proof of Fact 16.6.3, here is a basic and useful result. For atomic A and any $\sigma \in \mathcal{G}$,

$$\sigma A \in H^w \iff \mathcal{M}, \sigma \Vdash_{v_0} A. \quad (16.14)$$

Here is the argument, which is mostly just definition application. Let P be an n -place relation symbol and $t_1, \dots, t_n$ be members of the domain of the model $\mathcal{M}$, that is, parameters and pseudo-closed object terms that are positively generated in H^w . Also let $x_1, \dots, x_n$ be new variables, and let u be the variant of v_0 such that $u(x_i) = \llbracket t_i \rrbracket$ for $i = 1 \dots n$.

$$\begin{aligned} \sigma P(t_1, \dots, t_n) \in H^w & \\ \iff \langle t_1, \dots, t_n \rangle \in \mathcal{I}(P, \sigma) & \text{by definition of } \mathcal{I} \\ \iff \langle \llbracket t_1 \rrbracket, \dots, \llbracket t_n \rrbracket \rangle \in \mathcal{I}(P, \sigma) & \text{by Fact 16.6.2} \\ \iff \langle u(x_1), \dots, u(x_n) \rangle \in \mathcal{I}(P, \sigma) & \text{by definition of } u \\ \iff \mathcal{M}, \sigma \Vdash_u P(x_1, \dots, x_n) & \text{by Definition 16.1.4 part 1a} \\ \iff \mathcal{M}, \sigma \Vdash_{v_0} P(t_1, \dots, t_n) & \text{by Definition 16.1.4 part 1b} \end{aligned}$$

We now turn to Fact 16.6.3 itself. We show the following is the case, where Φ is any pseudo-closed formula, and σ is any prefix, including the witness prefix.

$$\begin{aligned} \text{if } \sigma \Phi \in H^w \text{ then } \mathcal{M}, \sigma \Vdash_{v_0} \Phi \\ \text{if } \sigma \neg\Phi \in H^w \text{ then } \mathcal{M}, \sigma \not\Vdash_{v_0} \Phi \end{aligned} \quad (\star\star\star)$$

Of course showing this involves a Complete Induction on the complexity of Z . We start with the base cases, which involve H-1 and H-6.

Atomic Cases Assume A is atomic. If $\sigma A \in H^w$ then $\mathcal{M}, \sigma \Vdash_{v_0} A$ by (16.14).

If $\sigma \neg A \in H^w$, then $\sigma A \notin H^w$ by H-1, so $\mathcal{M}, \sigma \not\Vdash_{v_0} A$ by (16.14) again, and so $\mathcal{M}, \sigma \Vdash_{v_0} \neg A$.

Equality Cases Assume t and u are parameters or object terms that are positively generated in H^w . By H-6 part a, $\sigma (t = t)$ is in H^w for every σ , so by (16.14), $\mathcal{M}, \sigma \Vdash_{v_0} (t = t)$.

Suppose $\mathcal{M}, \sigma \Vdash_{v_0} (t = u)$ and $\mathcal{M}, \tau \Vdash_{v_0} \Phi(t)$, where Φ is atomic. By (16.14) once more, both $\sigma (t = u)$ and $\tau \Phi(t)$ are in H^w . By H-6 part b, $\tau \Phi(u) \in H^w$, so $\mathcal{M}, \tau \Vdash_{v_0} \Phi(u)$ by (16.14) yet again.

Propositional, Modal, and Quantificational Cases This is the beginning of the inductive cases. For each case involving a formula, say Φ , the inductive hypothesis is that the result is true for formulas simpler than Φ . Note that H^w cannot contain any member of the form 2Ψ where Ψ is not atomic. So the non-atomic parts of all cases reduces to considering only ordinary prefixes. And so, the propositional cases, involving H-2, now are essentially the same as in Proposition 3.4.2. The modal cases, involving H-3, are as they were in Proposition 7.5.2. The quantifier cases, involving H-5, are as they were in Proposition 9.4.3. These arguments are not repeated here.

Predicate Abstraction Cases There are four cases, all involving H-7, and we only present the first. Assume that $\sigma \langle \lambda x. \Phi(x) \rangle (c) \in H^w$, and the induction hypothesis is that $(\star\star\star)$ is true for formulas that are simpler than this.

Since $\sigma \langle \lambda x. \Phi(x) \rangle (c) \in H^w$ by one part of Hintikka set condition H-7, $\sigma \Phi(c^\sigma) \in H^w$. By the induction hypothesis, $\mathcal{M}, \sigma \Vdash_{v_0} \Phi(c^\sigma)$. By Proposition 16.2.1, $\mathcal{M}, \sigma \Vdash_u \Phi(x)$, where u is the x -variant of v_0 such that $u(x) = \llbracket c^\sigma \rrbracket$. Note that $\llbracket c^\sigma \rrbracket = \mathcal{I}(c, \sigma)$. Then by Definition 14.6.2, part 10a, $\mathcal{M}, \sigma \Vdash_{v_0} \langle \lambda x. \Phi(x) \rangle (c)$.

The other three cases are similar but somewhat more complex, and are left to the reader.

We have now finished the proof of Proposition 16.6.1.

Exercises

Exercise 16.6.1 Give a proof for Fact 16.6.2.

16.7 Hintikka's Lemma, Second Pass

The version of Hintikka's Lemma in the form of Proposition 16.6.1 is not really what we want, because the model constructed is not normal, though because of the introduction of a witness world, it is a model as we define the term in this book. So we are close. Here is what we really need.

Proposition 16.7.1 (Hintikka's Lemma, Part 2) *Every first-order K Hintikka set with equality and predicate abstraction meeting the VN conditions (in which all prefixes are rooted at 1) is satisfiable in a varying domain K normal model in which terms might not designate.*

We have seen something like this before, in Proposition 11.7.2, when we were considering quantified modal logics with equality but had not yet seen predicate abstracts. The methodology now is exactly the same as it was then, and mostly what is needed here is just a review and an updating since we have more machinery now

than we did earlier. The idea of the proof was, and still is, to work with equivalence classes.

We directly follow the sequence of steps from the proof of Proposition 11.7.2. We start with the model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ that was constructed in the proof of Proposition 16.6.1, in which Hintikka set H^w is satisfiable. From it we create a new model $\overline{\mathcal{M}} = \langle \mathcal{G}, \mathcal{R}, \overline{\mathcal{D}}, \mathcal{I} \rangle$, closely related to $\mathcal{M}$, but normal. The set of possible worlds, and the accessibility relation remain the same in $\overline{\mathcal{M}}$ as in $\mathcal{M}$, but then things get interesting.

The domain of the model $\mathcal{M}$ is the set of all pseudo-closed terms that are parameters or that are positively generated in H^w (This is the same as $\mathcal{D}(2)$, the domain of the witness world.). As we did in Sect. 11.7, we define a relation on this model domain by calling model domain members t_1 and t_2 *equivalent* if $\mathcal{M}, \sigma \Vdash_{v_0} t_1 = t_2$ for some (any) $\sigma \in \mathcal{G}$, where v_0 maps each parameter to itself. As we did earlier, we denote this by $t_1 \sim t_2$. This is an equivalence relation, and the proof is much like it was earlier, so we skip the details now. For each member t of the domain of the model $\mathcal{M}$, let $\overline{t}$ be the equivalence class containing t . Using this, we can finish the construction of $\overline{\mathcal{M}}$.

Domain Function $\overline{\mathcal{D}}$ is the following mapping, assigning a set of equivalence classes to each possible world: $\overline{\mathcal{D}}(\sigma) = \{\overline{u} \mid u \in \mathcal{D}(\sigma)\}$.

Interpretation, Atomic Formulas For each k -place relation symbol R , $\overline{\mathcal{I}}(R, \sigma) = \{\langle \overline{t_1}, \dots, \overline{t_k} \rangle \mid \langle t_1, \dots, t_k \rangle \in \mathcal{I}(R, \sigma)\}$.

Interpretation, Constant Symbols For each non-rigid constant symbol c , $\overline{\mathcal{I}}(c, \sigma)$ is defined if $\mathcal{I}(c, \sigma)$ is defined, and $\overline{\mathcal{I}}(c, \sigma) = \overline{\mathcal{I}}(c, \sigma)$.

Interpretation, Function Symbols For each non-rigid k -place function symbol f , $\overline{\mathcal{I}}(f, \sigma)$ is defined if $\mathcal{I}(f, \sigma)$ is defined. $\langle \overline{t_1}, \dots, \overline{t_n} \rangle$ is in the domain of $\overline{\mathcal{I}}(f, \sigma)$ if $\langle t_1, \dots, t_n \rangle$ is in the domain of $\mathcal{I}(f, \sigma)$ and if it is, then $\overline{\mathcal{I}}(f, \sigma)(\overline{t_1}, \dots, \overline{t_k}) = \overline{\mathcal{I}}(f, \sigma)(t_1, \dots, t_k)$.

There are some problems with the definition above that should be addressed—specifically concerning the interpretations of atomic formulas and of function symbols. Suppose, for instance, that R is a relation symbol and we want to know if $\langle \overline{t_1}, \dots, \overline{t_k} \rangle \in \overline{\mathcal{I}}(R, \sigma)$. According to the definition, this will be the case if $\langle t_1, \dots, t_k \rangle \in \mathcal{I}(R, \sigma)$. But it could easily be the case that $\langle \overline{t_1}, \dots, \overline{t_k} \rangle$ and $\langle \overline{u_1}, \dots, \overline{u_k} \rangle$ are the same while $\langle t_1, \dots, t_k \rangle$ and $\langle u_1, \dots, u_k \rangle$ might not be. Does this lead to trouble? The answer is no, and the reasons why were discussed in Sect. 11.7 (see the material in that section following **Normal Model Construction**). Similar issues come up concerning the interpretation of function symbols, and we leave them to you.

Generally speaking, the proofs from Sect. 11.7 continue to work, with minor modifications. In particular $\overline{\mathcal{M}}$ is a normal model, and behaves the same as $\mathcal{M}$ in the following sense where, for each valuation v in $\mathcal{M}$, the valuation $\overline{v}$ in $\overline{\mathcal{M}}$ is the one such that $\overline{v}(x) = v(x)$.

Model Equivalence For each formula Ψ , each $\sigma \in \mathcal{G}$, and each valuation v :

$$\mathcal{M}, \sigma \Vdash_v \Psi \iff \overline{\mathcal{M}}, \sigma \Vdash_{\bar{v}} \Psi.$$

This is proved by complete induction on the complexity of Ψ . Most of the cases are essentially as they were in Sect. 11.7 and are not repeated here. The cases not covered in the earlier chapter are those of predicate abstraction and, as usual, there are four: constants or function symbols, and positive or negative occurrences. We present a single case, constant symbol, positive occurrence, as representative. In the proof of it we make use of two items from Sect. 11.7. These still apply in the present, richer setting. Here is the one induction step we treat in detail.

Predicate Abstraction Suppose Ψ is $\langle \lambda x. \Phi \rangle (c)$ and the result is known for simpler formulas. Here are the steps. The action really takes place in the side conditions, and discussion of them follows.

$$\mathcal{M}, \sigma \Vdash_v \langle \lambda x. \Phi \rangle (c) \iff \mathcal{M}, \sigma \Vdash_w \Phi \text{ where } w(x) = \mathcal{I}(c, \sigma) \quad (16.15)$$

$$\iff \overline{\mathcal{M}}, \sigma \Vdash_{\bar{w}} \Phi \text{ where } \bar{w}(x) = \overline{\mathcal{I}}(c, \sigma) \quad (16.16)$$

$$\iff \overline{\mathcal{M}}, \sigma \Vdash_{\bar{v}} \langle \lambda x. \Phi \rangle (c) \quad (16.17)$$

In (16.15), w is the x variant of v such that $w(x) = \mathcal{I}(c, \sigma)$. Likewise in (16.16), $\bar{w}$ is the x variant of $\bar{v}$ such that $\bar{w}(x) = \overline{\mathcal{I}}(c, \sigma)$. Then the equivalences in (16.15), and between (16.16) and (16.17) are justified by Definition 14.6.2 part 10a. It is the equivalence between (16.15) and (16.16) that needs more discussion.

First, assume that $\mathcal{M}, \sigma \Vdash_w \Phi$ where w is an x variant of v such that $w(x) = \mathcal{I}(c, \sigma)$. By the induction hypothesis, since we have $\mathcal{M}, \sigma \Vdash_w \Phi$, we have $\overline{\mathcal{M}}, \sigma \Vdash_{\bar{w}} \Phi$. Further, using *Valuation Fact One* from Sect. 11.7, since w is an x variant of v then $\bar{w}$ is an x variant of $\bar{v}$. And, $\bar{w}(x) = w(x) = \mathcal{I}(c, \sigma) = \overline{\mathcal{I}}(c, \sigma)$.

Second, assume that $\overline{\mathcal{M}}, \sigma \Vdash_{\bar{w}} \Phi$ where $\bar{w}$ is an x variant of $\bar{v}$ such that $\bar{w}(x) = \overline{\mathcal{I}}(c, \sigma)$. Then using *Valuation Fact Two* from Sect. 11.7, there must be a valuation w in $\mathcal{M}$ such that $\bar{w}$ is the equivalence class containing w , and w is an x variant of v . By the induction hypothesis, $\mathcal{M}, \sigma \Vdash_w \Phi$.

We have now ended our sketch of the **Model Equivalence** result, thus concluding our sketch of the proof of Proposition 16.7.1.

16.8 Completeness for K (VN Version)

Things now proceed in a familiar way. We need a systematic tableau construction algorithm that takes predicate abstraction into account, and uses the VN conditions. (Or a maximal consistent set construction would also serve.) We have seen simpler versions of such algorithms in previous chapters, whenever a tableau completeness theorem was being proved. The details for what is needed now are mildly fussy,

and we omit them. You should convince yourselves that it is possible to formulate a systematic tableau construction algorithm, and that should be enough for us to get on with things. The important point is that, if such an algorithm does not produce a proof, then the set of prefixed formulas on any open branch will be a Hintikka set according to Definition 16.4.1. Note: this does not involve witness prefixes, which play no role in tableau construction, but come in during the proof of Hintikka's Lemma. Summarizing, we have the following, which should look familiar from earlier chapters.

Let Φ be a closed formula that does not have a tableau proof. A systematic tableau construction beginning with $1 \neg\Phi$ cannot close, and hence will generate at least one open branch. The prefixed formulas on such an open branch are a Hintikka set H . Summarizing: if Φ is not provable, there is a Hintikka set H containing $1 \neg\Phi$.

Once we have Hintikka set H , we can extend it to another, H^w , involving a witness prefix, Sect. 16.5. Then the Hintikka set construction of Proposition 16.6.1 gives us a model in which the possible worlds are the prefixes, and φ fails at world 1. Note that it is really a model in the sense we have been using the word: terms that designate will designate an existent in some possible world. This is because of the presence of the witness world in the model. Roughly, to be a possible existent is to be an actual existent in some possible world. But this model will not be a normal one.

Finally the model we have can be converted to a normal model, Proposition 16.7.1, and formula φ will fail in that, once again at possible world 1. Tableau completeness has been established.

Exercises

Exercise 16.8.1 This is more of a project than a simple exercise. Prove completeness for the *CN* version of **K** tableaux. This is considerably simpler than the *VN* version we presented. For one thing, witness worlds are not needed since all worlds have the same domain. Indeed, everything is less complicated than what you saw in this chapter.

Part VI

Applications

We have now finished presenting the primary formal machinery used in this book. We have modal operators, quantifiers allowing both constant and varying domains, equality, predicate abstracts, non-rigid constant and function symbols that always designate, and those that might not. Besides semantical machinery we also have proof systems, soundness and completeness arguments, and so on. Along the way we discussed some of the applications of this machinery, in modeling concepts of philosophical interest. We are now finished with the presentation of the basic formal machinery, but we are not finished with the examination of their applications.

Chapter 17

Equality and Predicate Abstraction



17.1 The Role of Scope

We discussed equality at length in Chap. 11, but this was before we introduced the machinery of predicate abstraction. Recall that a model is *normal* if the relation symbol “=” is interpreted to be the equality relation on the domain of the model. The relation is thus the same from world to world. Now it is time to see how equality and predicate abstraction interact. While the basics of what we have to say applies to non-rigid terms generally, things are most easily understood if function symbols are not present, and that is all we discuss formally in this section. Our setting throughout this section can be assumed to be the modal logic **K**, with varying domains, allowing terms that may not designate, that is, **K** with the VN conditions. This is the most general setting we have, since all other setups are the result of putting further restrictions on this one.

The first, unsurprising, fact is the following, which is almost immediate and so we omit a proof. In it we use the abbreviation from Definition 14.1.3.

Proposition 17.1.1 *In a normal model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ meeting the VN conditions, for constant symbols c and d ,*

$$\mathcal{M}, \Gamma \Vdash_v \langle \lambda w, z. (w = z) \rangle (c, d) \iff \text{both } c \text{ and } d \text{ designate at } \Gamma, \\ \text{and designate the same thing there.}$$

Things become more interesting when we start looking at the interplay between equality and necessity when predicate abstracts are involved. All of the following are plausible abstractions arising from $\Box(x = y)$, where c and d are non-rigid constant symbols. But by no means do they all have equivalent behavior.

$$\text{E-1 } \langle \lambda x, y. \Box(x = y) \rangle (c, d)$$

$$\text{E-2 } \Box \langle \lambda x, y. (x = y) \rangle (c, d)$$

E-3 $\langle \lambda x. \Box \langle \lambda y. (x = y) \rangle (d) \rangle (c)$

E-4 $\langle \lambda y. \Box \langle \lambda x. (x = y) \rangle (c) \rangle (d)$

We recall from Chap. 11 that

$$(x = y) \supset \Box(x = y) \quad (17.1)$$

is valid, where x and y are free variables. But E-1 through E-4 give us not one, but four, plausible analogs of (17.1). In the following, c and d are again non-rigid constant symbols.

N-1 $\langle \lambda x, y. (x = y) \rangle (c, d) \supset \langle \lambda x, y. \Box(x = y) \rangle (c, d)$

N-2 $\langle \lambda x, y. (x = y) \rangle (c, d) \supset \Box \langle \lambda x, y. (x = y) \rangle (c, d)$

N-3 $\langle \lambda x, y. (x = y) \rangle (c, d) \supset \langle \lambda x. \Box \langle \lambda y. (x = y) \rangle (d) \rangle (c)$

N-4 $\langle \lambda x, y. (x = y) \rangle (c, d) \supset \langle \lambda y. \Box \langle \lambda x. (x = y) \rangle (c) \rangle (d)$

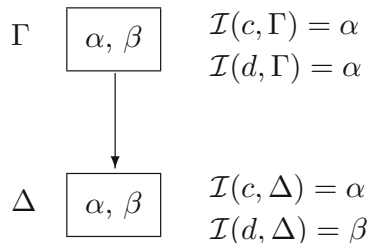
Additional possibilities arise if more than one necessity operator is allowed, and it is reasonable to ask how complex all this can get. Perhaps more significantly, it turns out that no two of these are equivalent. In a sense, it is version N-1 that is the closest analog of (17.1), as the following example suggests.

Example 17.1.2 The formula (N-1) is valid.

Suppose $\mathcal{M}, \Gamma \Vdash_v \langle \lambda x, y. (x = y) \rangle (c, d)$, where $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ is a normal model. Then both c and d designate and $\mathcal{I}(c, \Gamma) = \mathcal{I}(d, \Gamma)$, by Proposition 17.1.1. We must show that we also have $\mathcal{M}, \Gamma \Vdash_v \langle \lambda x, y. \Box(x = y) \rangle (c, d)$. Following through the definitions, this means we must show we have $\mathcal{M}, \Gamma \Vdash_{v'} \Box(x = y)$, where v' agrees with v on all variables except x and y , and $v'(x) = \mathcal{I}(c, \Gamma)$, and $v'(y) = \mathcal{I}(d, \Gamma)$. To show this, let Δ be an arbitrary world accessible from Γ . We must show we have $\mathcal{M}, \Delta \Vdash_{v'} (x = y)$. But this is the case since $v'(x) = \mathcal{I}(c, \Gamma) = \mathcal{I}(d, \Gamma) = v'(y)$.

The example above can be read as: consider the objects that c and d denote; if those objects are equal, they are necessarily equal. This is not really about non-rigid denotation—it is about the things denoted. Equal objects are equal, no matter what. More colloquially, it can be read: if it is true *of* c and d that they are equal, then it is also true *of* them that they are necessarily equal.

Example 17.1.3 The formula (N-2) is not valid. This is important, since (N-1) and (N-2) are easily confused. Consider the following schematically presented model $\mathcal{M}$ (which, incidentally, is constant domain).



We have $\mathcal{M}, \Gamma \Vdash_v \langle \lambda x, y.(x = y) \rangle(c, d)$ (for any v) because this reduces to $\mathcal{I}(c, \Gamma) = \mathcal{I}(d, \Gamma)$ and this is correct since both have the value α .

Next, if we had $\mathcal{M}, \Gamma \Vdash_v \Box \langle \lambda x, y.(x = y) \rangle(c, d)$, we would also have $\mathcal{M}, \Delta \Vdash_v \langle \lambda x, y.(x = y) \rangle(c, d)$, which reduces to $\mathcal{I}(c, \Delta) = \mathcal{I}(d, \Delta)$. But this is not the case since $\mathcal{I}(c, \Delta) = \alpha$ and $\mathcal{I}(d, \Delta) = \beta$, which are different. Thus $\langle \lambda x, y.(x = y) \rangle(c, d) \supset \Box \langle \lambda x, y.(x = y) \rangle(c, d)$ fails in this model.

This example can be read as asserting: if c and d designate the same object, then necessarily they designate the same object. Read this way, we should not be surprised at invalidity. Just because c and d designate the same thing today doesn't mean they must do so tomorrow. Again, read colloquially, we have the invalidity of: if it is true *that* c and d are equal, then it is necessarily true *that* c and d are equal.

The two examples above have significance for the morning star/evening star puzzle. Let m and e be the non-rigid designators “the morning star,” and “the evening star,” respectively, and read $\Box$ as “the ancients knew.” There is no doubt that in the actual world, the morning star is the evening star. Unabbreviated,

$$\langle \lambda x, y.(x = y) \rangle(m, e). \quad (17.2)$$

Now by Example 17.1.2, the sentence

$$\langle \lambda x, y.(x = y) \rangle(m, e) \supset \langle \lambda x, y.\Box(x = y) \rangle(m, e) \quad (17.3)$$

is valid. It follows from (17.2) and (17.3) that

$$\langle \lambda x, y.\Box(x = y) \rangle(m, e) \quad (17.4)$$

is true in the actual world. This is, in fact, correct. The phrases “morning star” and “evening star” designate the same object, and the ancients certainly knew *of* any object that it is identical with itself.

If the sentence

$$\langle \lambda x, y.(x = y) \rangle(m, e) \supset \Box \langle \lambda x, y.(x = y) \rangle(m, e) \quad (17.5)$$

were also valid, then from the true (17.2) we could conclude

$$\Box \langle \lambda x, y.(x = y) \rangle(m, e). \quad (17.6)$$

But Example 17.1.3 shows that the sentence (17.5) is simply not a valid one. So the problematic conclusion (17.6), which can be read “The ancients knew *that* the morning star and the evening star are equal,” cannot be drawn.

17.2 More Examples

Earlier in this book we gave various and somewhat complex formulas to be proved using tableaux. At that point they were just formal exercises. Now we place them in anecdotal contexts, illustrating how predicate abstracts allow us to disambiguate what could otherwise be confusing situations.

Example 17.2.1 The sentence

$$\left[\langle \lambda x. \Diamond \langle \lambda y. (x = y) \rangle (p) \rangle (a) \wedge \langle \lambda x. \Box S(x) \rangle (a) \right] \supset \\ \Diamond \langle \lambda x. S(x) \rangle (p)$$

is valid. We gave a tableau proof of it in Example 15.5.5, in **K** using the *VA* conditions, and in Exercise 15.5.5 we asked you to verify that our proof also worked using the *VN* conditions. A concrete instance of this sentence should sound more intelligible to you. Note our careful usage of “of” and “that” in what follows. We give the modal operators a temporal reading: $\Box X$ is read, “ X will always be the case in the future,” and $\Diamond X$ is read, “ X will be the case some time in the future.” Suppose the possible world in which we evaluate the sentence is the world of 1850. And finally, suppose that a is “Abraham Lincoln,” p is “the President of the United States,” and $S(x)$ is “ x sees the start of the American Civil War.” Under this reading,

$$\langle \lambda x. \Diamond \langle \lambda y. (x = y) \rangle (p) \rangle (a)$$

says, “It is true, of Abraham Lincoln, that he will be President of the United States.” Likewise,

$$\langle \lambda x. \Box S(x) \rangle (a)$$

reads, “It is true of Abraham Lincoln that he sees the start of the Civil War.” The conclusion we draw is

$$\Diamond \langle \lambda x. S(x) \rangle (p)$$

and this reads “It is true that at some time in the future the President of the United States sees the start of the Civil War.”

As we noted above, we provided a tableau proof of this in Example 15.5.5. We leave it to you as an exercise to verify its validity by a direct semantical argument.

Example 17.2.2 Suppose you tell me, “Today is Bastille day,” and I respond, “I didn’t know that. I know that Bastille day is always July 14, but I didn’t know today was July 14.” Let us see how we formalize this small dialogue.

We give $\Box X$ the epistemic reading “I know that X .” We let b be “Bastille day,” t be “today,” and j be “July 14.” These three constants are non-rigid, for there is

no assurance that they all will designate the same day every time, and consequently, their designation can vary with different states of my knowledge.

Now, you tell me that today is Bastille day,

$$\langle \lambda x, y.(x = y) \rangle(t, b).$$

From this it follows, by Example 17.1.2, that

$$\langle \lambda x, y.\Box(x = y) \rangle(t, b),$$

i.e., I know, *of* today and *of* Bastille day, that they are the same. But this is trivial. They are in fact the same, and I know of each object that it is self-identical.

I claimed to know *that* Bastille day is always July 14, that is, I asserted

$$\Box \langle \lambda x, y.(x = y) \rangle(b, j).$$

But I also said I did not know, of today, that it was July 14, that is,

$$\langle \lambda x.\neg\Box \langle \lambda y.(x = y) \rangle(j) \rangle(t).$$

This purported to explain my not knowing, of today, that it was Bastille day, that is,

$$\langle \lambda x.\neg\Box \langle \lambda y.(x = y) \rangle(b) \rangle(t).$$

Now, in fact, the sentence

$$[\Box \langle \lambda x, y.(x = y) \rangle(b, j) \wedge \langle \lambda x.\neg\Box \langle \lambda y.(x = y) \rangle(j) \rangle(t)] \supset \\ \langle \lambda x.\neg\Box \langle \lambda y.(x = y) \rangle(b) \rangle(t)$$

is valid. We leave it to you to verify this. Alternatively, it too can be proved using tableau rules. Since quantifiers are not involved, it doesn't matter whether constant or varying domains are used. Whether constant symbols always designate or might not, the second option is the more general. So verify this semantically or using tableaux, under the *CN* assumptions.

Example 17.2.3 In 1556 Charles V, Holy Roman Emperor, resigned his offices and spent the remainder of his life near the monastery of San Yuste in Spain repairing clocks. One can imagine Charles, around 1550, planning these events and musing “Someday the Emperor won’t be the Emperor.”

We give $\Box$ a temporal reading as in Example 17.2.1. Also, let c be “Charles V,” and e be “Holy Roman Emperor.”

Charles is certainly not asserting $\langle \lambda x, y.\Diamond\neg(x = y) \rangle(c, e)$, which says *of* himself, Charles, and himself, the Emperor, that they will at some time be different individuals. For in that case he would be uttering a contradiction. After all, at the time Charles spoke, he was the Emperor, so it was true that $\langle \lambda x, y.x = y \rangle(c, e)$,

and the sentence $\langle \lambda x, y. x = y \rangle(c, e) \supset \neg \langle \lambda x, y. \Diamond \neg(x = y) \rangle(c, e)$, is valid. Rather, Charles is asserting what is formalized by $\langle \lambda x. \Diamond \langle \lambda y. \neg(x = y) \rangle(e) \rangle(c)$, i.e., it is true, of Charles, that he has the property of someday being different than the Emperor. In this case we do not have a validity to deal with—there is nothing logically certain about Charles' assertion. But the sentence is *satisfiable*—there is a way of making it true in a model. It could happen.

Exercises

Exercise 17.2.1 Determine the status of each of the following sentences. That is, either show validity, or produce a counter-example for:

1. Sentence N-3,
2. Sentence N-4.

Exercise 17.2.2 Show the validity of the sentence of Example 17.2.1 using the *VN* assumption.

Exercise 17.2.3 Show the validity of the sentence of Example 17.2.2 using *CN* assumption.

Exercise 17.2.4 Give a tableau proof of the sentence at the end of Example 17.2.2. What tableau conditions are needed. That is, can it be proved assuming the *VN* conditions, or is something stronger needed?

Exercise 17.2.5 Finish Example 17.2.3 and produce a model in which the sentences

$$\begin{aligned} &\langle \lambda x, y. x = y \rangle(c, e) \\ &\neg \langle \lambda x, y. \Diamond \neg(x = y) \rangle(c, e) \\ &\langle \lambda x. \Diamond \langle \lambda y. \neg(x = y) \rangle(e) \rangle(c) \end{aligned}$$

are all true at some possible world.

Exercise 17.2.6 Choose any two of the sentences E-1, E-2, E-3, E-4. Construct a model in which there is a possible world where your first choice is true while your second is false.

Chapter 18

Designation



There is a natural confusion between existence and designation, but these are really orthogonal issues. Terms designate; objects exist. For instance the phrase “the first President of the United States” designates George Washington, though thinking temporally, the person being designated is no longer with us—the person designated does not exist, though he once did. The nonexistent George Washington is designated *now* by the phrase. On the other hand the phrase, “the present King of France,” does not designate anybody now, living or dead, though at certain past instances it did designate.

In our formal treatment we have allowed terms to designate nonexistent objects at worlds of varying domain models. More properly, a term may designate, at a world of a varying domain model, an object not in the domain of that world, though it must be in the domain of *some* possible world, as we are treating things. It is the purpose of this chapter to elaborate on connections between designation and existence. There are some, but they are most decidedly different things.

While definite descriptions are the classic examples of things that may fail to designate, we do not introduce them formally until Chap. 20. In the meantime we can, and will, consider them as motivational examples, as we did above.

18.1 Designation and Existence

Designation is an inherently semantic notion. Nonetheless, a syntactic counterpart is available for it. We begin this section by introducing a “designation property,” then we investigate its relationships with the existence property from Definition 15.4.3, and more generally with the material in Chap. 12.

Definition 18.1.1 (Designates) **D** abbreviates $\langle \lambda x.x = x \rangle$, and so **D**(*t*) abbreviates $\langle \lambda x.x = x \rangle(t)$.

Think of $\mathbf{D}(t)$ as intended to assert that t designates. The choice of the formula $x = x$ is a little arbitrary. All we really need is a formula that cannot be falsified; $P(x) \vee \neg P(x)$ would do as well, though it requires us to choose a relation symbol P , and this has an even greater element of arbitrariness about it.

Earlier we gave a characterization of when a constant or function symbol designated, at a world in a model, Definition 14.6.1. A constant symbol designated at a world if the interpretation function of the model was defined for that constant symbol, at that world. Similarly for function symbols. These are semantic notions. The key reason we now introduce $\mathbf{D}$ is that it allows us to move these meta-level notions into the object language itself. Here is the formal statement of this fact.

Proposition 18.1.2 *Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a varying domain normal model, in which terms might not designate, and let c be a constant symbol and f be an n place function symbol.*

1. $\mathcal{M}, \Gamma \Vdash_v \mathbf{D}(c)$ if and only if c designates at Γ in $\mathcal{M}$.
2. $\mathcal{M}, \Gamma \Vdash_v \mathbf{D}(f(x_1, \dots, x_n))$ if and only if f designates at Γ in $\mathcal{M}$ and $\langle v(x_1), \dots, v(x_n) \rangle$ is in the domain of $\mathcal{I}(f, \Gamma)$.

Proof This is quite simple, and follows directly from our definitions. We present case 1; case 2 is similar. If c designates at Γ in $\mathcal{M}$, $\mathcal{I}(c, \Gamma)$ is defined, and since we are assuming $\mathcal{M}$ is a normal model, it follows from Definition 14.6.2 that $\mathcal{M}, \Gamma \Vdash_v \langle \lambda x. x = x \rangle(c)$. On the other hand if c does not designate at Γ , by Definition 14.6.2 again, $\mathcal{M}, \Gamma \Vdash_v \langle \lambda x. \Phi \rangle(c)$ is false no matter what Φ may be, so in particular, $\mathcal{M}, \Gamma \Vdash_v \langle \lambda x. x = x \rangle(c)$ is false. $\square$

Example 18.1.3 Although It will be seen that $\neg \langle \lambda x. \Phi(x) \rangle(c) \supset \langle \lambda x. \neg \Phi(x) \rangle(c)$ is not valid if c does not designate, we do have the validity of the following.

$$\mathbf{D}(c) \supset [\neg \langle \lambda x. \Phi(x) \rangle(c) \supset \langle \lambda x. \neg \Phi(x) \rangle(c)]$$

Here is a tableau proof using **K** rules with the *VN* assumptions.

- 1 $\neg \{ \mathbf{D}(c) \supset [\neg \langle \lambda x. \Phi(x) \rangle(c) \supset \langle \lambda x. \neg \Phi(x) \rangle(c)] \}$ 1.
- 1 $\mathbf{D}(c)$ 2.
- 1 $\neg [\neg \langle \lambda x. \Phi(x) \rangle(c) \supset \langle \lambda x. \neg \Phi(x) \rangle(c)]$ 3.
- 1 $\neg \langle \lambda x. \Phi(x) \rangle(c)$ 4.
- 1 $\neg \langle \lambda x. \neg \Phi(x) \rangle(c)$ 5.
- 1 $\langle \lambda x. x = x \rangle(c)$ 6.
- 1 $c^1 = c^1$ 7.
- 1 $\neg \Phi(c^1)$ 8.
- 1 $\neg \neg \Phi(c^1)$ 9.

Items 2 and 3 are from 1 by a Conjunctive Rule, as are 4 and 5 from 3; 6 is simply 2 rewritten, from which 7 follows by the Positive Abstraction Rule; now 8 is from 4 and 9 is from 5 by the Negative Abstraction Rule, and we have closure at this point.

Earlier, in Definition 15.4.3, we introduced an existence predicate **E**. We repeat this here for convenience. **E** abbreviates $\langle \lambda x. \mathbf{E}(x) \rangle$ or, fully unabbreviated, $\langle \lambda x. (\exists y)(y = x) \rangle$. In a similar way we now define a *nonexistence* predicate:

Definition 18.1.4 (Non-Existence Abstract) By $\bar{\mathbf{E}}$ we mean the predicate abstract $\langle \lambda x. \neg \mathbf{E}(x) \rangle$, or unabbreviated, $\langle \lambda x. \neg (\exists y)(y = x) \rangle$.

The distinction between having the non-existence property and not having the existence property is worth a bit of discussion, which we postpone until after the Proposition below. For now it is enough to say, they are not the same.

Proposition 18.1.5 *For each constant symbol c and each n -ary function symbol f , the following are valid in all varying domain normal models in which terms might not designate, that is, in VN models.*

1. $\mathbf{D}(c) \equiv [\mathbf{E}(c) \vee \bar{\mathbf{E}}(c)]$
2. $\mathbf{D}(f(x_1, \dots, x_n)) \equiv [\mathbf{E}(f(x_1, \dots, x_n)) \vee \bar{\mathbf{E}}(f(x_1, \dots, x_n))]$

Proof We show part 1; part 2 is similar. If c does not designate (at some particular world), no predicate abstracts correctly apply to c ; and thus in particular, both $\mathbf{E}(c)$ and $\bar{\mathbf{E}}(c)$ are false.

The other direction is only a little more work. Suppose c does designate at a world of the model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$, say $\mathcal{M}, \Gamma \Vdash_v \mathbf{D}(c)$. Then $\mathcal{I}(c, \Gamma)$ is defined. Now, either $\mathcal{I}(c, \Gamma)$ is in $\mathcal{D}(\Gamma)$ or it is not. If it is, $\mathcal{M}, \Gamma \Vdash_v \mathbf{E}(c)$ is true, and if it is not, $\mathcal{M}, \Gamma \Vdash_v \bar{\mathbf{E}}(c)$ is true. Either way, we have $\mathbf{E}(c) \vee \bar{\mathbf{E}}(c)$. $\square$

If a term t designates at a world, it follows from this Proposition that $\mathbf{E}(t) \vee \bar{\mathbf{E}}(t)$, and hence $\bar{\mathbf{E}}(t) \equiv \neg \mathbf{E}(t)$. Then if all terms always designate, a special nonexistence predicate abstract is redundant—negating the existence predicate abstract suffices. If non-designation can happen, things are different. If t does not designate at a particular world, *both* $\mathbf{E}(t)$ and $\bar{\mathbf{E}}(t)$ must be false there, and we do not have $\bar{\mathbf{E}}(t) \equiv \neg \mathbf{E}(t)$! Stating this in an unabbreviated version, if t does not designate at a world, $\langle \lambda x. \neg (\exists y)(y = x) \rangle(t)$ and $\neg \langle \lambda x. (\exists y)(y = x) \rangle(t)$ are not equivalent at that world.

In earlier chapters the machinery of predicate abstraction played a significant role when dealing with modal notions, but its effects were invisible where the classical connectives and quantifiers were concerned. Now that non-designation is allowed, predicate abstraction is visible even at the non-modal level. As we have just seen, negation inside and negation outside a predicate abstract can be different. This fact was noted long ago by Russell (1905) in his treatment of definite descriptions, which discusses classic examples of possible non-designation.

Exercises

Exercise 18.1.1 Analogous to the nonexistence predicate introduced above, one could also introduce a non-designation predicate. Do so, then explain why the result is useless.

Exercise 18.1.2 Which of the following are valid? Assume c is a non-rigid constant symbol and the semantics meets the VN conditions.

1. $\langle \lambda x. \neg \Phi \rangle(c) \supset \neg \langle \lambda x. \Phi \rangle(c)$.
2. $\neg \langle \lambda x. \Phi \rangle(c) \supset \langle \lambda x. \neg \Phi \rangle(c)$.
3. $\langle \lambda x. \Phi \supset \psi \rangle(c) \supset [\langle \lambda x. \Phi \rangle(c) \supset \langle \lambda x. \psi \rangle(c)]$.
4. $[\langle \lambda x. \Phi \rangle(c) \supset \langle \lambda x. \psi \rangle(c)] \supset \langle \lambda x. \Phi \supset \psi \rangle(c)$.

Exercise 18.1.3 Give a tableau proof of the following (which is the converse of Exercise 18.1.3):

$$[\neg \langle \lambda x. \Phi(x) \rangle(c) \supset \langle \lambda x. \neg \Phi(x) \rangle(c)] \supset \mathbf{D}(c)$$

Exercise 18.1.4 Give a tableau proof of $\mathbf{D}(c) \equiv [\mathbf{E}(c) \vee \overline{\mathbf{E}}(c)]$.

18.2 Existence and Designation

Throughout this section only varying domain models will be considered. The use of actualist quantification is critical for the points we wish to make.

One of the interesting results of our treatment of varying domain semantics is that everything exists; however, as Quine put it, we quibble over cases. The “ x exists” formula, defined as

$$\mathbf{E}(x) \text{ or } (\exists y)(y = x), \quad (18.1)$$

is an open sentence with a free variable. It is important to distinguish (18.1) from

$$(\forall x)\mathbf{E}(x) \text{ or } (\forall x)(\exists y)(y = x), \quad (18.2)$$

which is a (closed) sentence and, importantly, a logical truth: Everything exists. The open formula (18.1) is far from a logical truth. To the contrary, it is (speaking loosely) true of some things and false of others. Bear in mind our use of free variables. An object that is not in the domain of the actual world can be the value of a free variable. Such an object is a nonexistent object in the actual world. And if such an object is taken to be the value of the free variable x , then (18.1) is not true of it: there is nothing in the domain of this world with which it is identical.

By contrast with (18.1), the formula

$$x = x, \quad (18.3)$$

which is also an open sentence with one free variable, is true of every existent and every nonexistent. Formula (18.3) is true for any object that exists in any possible world. We might, if we wanted to keep the Meinongian terminology alive, take the predicate abstracted from (18.3), $(\lambda x.x = x)$, to be the “has being” predicate. But we prefer to think of it as saying that a term designates.

Although everything exists, we can say, of a given object, that *it* exists, and speak either truly or falsely; and we can say, of a given object, that *it* does not exist, and speak either truly or falsely. There are no paradoxical consequences to be drawn from this. We cannot say everything has being: if we attempt to do so, we come out with the slightly different

$$(\forall x)(x = x), \quad (18.4)$$

which is true of everything in the range of the quantifier, i.e., everything that exists. What we can say is that every instance of (18.3) is true, and so, we cannot truthfully deny *of* a given object that *it* has being.

Does this mean that we are saddled with the Meinongian thesis that the round-square has being or that the least prime number between 19 and 23 has being? No. We are not committed to Meinong’s comprehension schema. The meaning of a singular term need not be identified with the thing it designates, and so, a singular term can be meaningful even if it doesn’t designate anything. Furthermore, there is no need to suppose that inconsistent singular terms need to designate anything to be meaningful. The singular term “the round-square” does not designate in any possible world. And in the case of “the round-square,” we can coherently deny that the round-square exists, even that the round-square is identical with itself; what we cannot do, since the term fails to designate, is deny *of* the round-square that it exists, or again, deny *of* the round-square that it has being.

In Chap. 20 we give a formal version of these ideas, but until definite descriptions have been discussed in detail, such a treatment must wait. In the meantime, we continue with our informal exposition, preparing the way for the formalizations that follow.

Let us give a more concrete example of the distinctions we have been drawing. Suppose we are working within a temporal framework, so that our frame contains a set of possible worlds with a temporal accessibility relation. In the present temporal world, George Washington doesn’t exist. This is not to say that *something* fails to exist, for the point of saying that he doesn’t exist is that there is nothing identical with him, i.e. none of the things that exist is identical with him. The expression “George Washington” *designates*, but fails to *designate an existent*. It designates an object that exists in an earlier temporal world. Recall our temporal operators **P** and **F** (*It was the case that*, *It will be the case that*). Now, George Washington purportedly

had wooden teeth. We use the following abbreviations: g for “George Washington” and $T(x)$ for “ x has wooden teeth.” So, we have the following truth:

$$\mathbf{P}(\lambda x.T(x))(g) \quad (18.5)$$

Kripke has argued that a proper name like “George Washington” is a rigid designator when we interpret $\Box$ as metaphysical necessity. There is no *a priori* reason to think that because an expression is rigid under one interpretation of $\Box$ that it is therefore rigid under all; but the sort of arguments put forward for the claim that proper names are rigid in the context of metaphysical necessity appear to be equally appropriate for the case of temporal necessity. The salient difference between the alethic and the temporal case is that George Washington does not exist *now*. As a result, we have a rigid designator, “George Washington”, that rigidly designates a nonexistent and further, one whose baptismal ceremony took place in another (earlier) world in the model, but not the actual (present) one. This further underscores the correctness of separating out the issues of rigidity and existence.

It is reasonable to argue that “George Washington” is temporally a rigid designator, and this implies that (18.5) is logically indistinguishable from

$$\langle \lambda x.\mathbf{PT}(x) \rangle(g). \quad (18.6)$$

George Washington doesn’t exist—certainly not in this temporal world, although he did exist in a previous temporal world. So, the upshot is that rigid designators can be introduced for nonexistents. In another temporal world, the man exists and is so baptized; we intend to maintain that reference even though the man no longer exists. We can speak about him even though he doesn’t exist.

So we have the proper name “George Washington” and even though we have true things to say about George Washington, he doesn’t exist. In particular, even if it turns out to be true that he, i.e., George Washington, had wooden teeth, it does not follow that *something* or *someone*—where these are understood as existential quantifications—had wooden teeth. From (18.6), then, we cannot infer

$$(\exists y)\langle \lambda x.\mathbf{PT}(x) \rangle(y). \quad (18.7)$$

And the reason is quite clear: George Washington does not exist now, and so existential generalization does not apply.

Contrast the expressions “George Washington” and “George Washington’s eldest son.” Not only does this latter expression fail to designate in this temporal world (as does “George Washington”), but it fails to designate in every temporal world. There is no temporal world in which George Washington has an eldest son. From a temporal world perspective, then, there is no “possible world” in which George Washington’s eldest son exists. We have here a somewhat different situation from that of George Washington. George Washington, we say, doesn’t exist but did. There is some possible world in which we find him in the domain. George Washington’s eldest son doesn’t exist and can’t. His existence is impossible because of the

semantics chosen in the model, so “George Washington’s eldest son” fails to designate. (This is the formal way of saying that he lacks being, in Meinongian terms.) We can, of course, deny his existence. But here we must be careful. Suppose we let w be “George Washington’s eldest son.” This is a constant which fails to designate anything that exists in the current (or any other) temporal world. Since it fails to designate in any world, it fails to designate anything that has a property; so, in particular, it fails to designate anything that has the nonexistence property. This means that we cannot express the claim

$$\text{George Washington's eldest son does not exist.} \quad (18.8)$$

as

$$\langle \lambda x. \neg(\exists y)(y = x) \rangle(w). \quad (18.9)$$

Rather, we must do so like this:

$$\neg \langle \lambda x. (\exists y)(y = x) \rangle(w). \quad (18.10)$$

It is not, then, that w fails to exist; rather, the term fails to designate! We could just as well have used our existence formula $\mathbf{E}(x)$ to represent (18.9) and (18.10), respectively, as

$$\langle \lambda x. \neg \mathbf{E}(x) \rangle(w), \text{ or equivalently, } \bar{\mathbf{E}}(w), \quad (18.11)$$

which is false, since it asserts that w has a property, nonexistence, and

$$\neg \langle \lambda x. \mathbf{E}(x) \rangle(w), \text{ or equivalently, } \neg \mathbf{E}(w), \quad (18.12)$$

which is true. Predicate abstraction notation is crucial here. Without it, the symbolization of (18.8) would be the ambiguous

$$\neg \mathbf{E}(w). \quad (18.13)$$

The most radical type of designation failure is an inconsistent description, e.g., “the round square” or “the smallest prime number between 19 and 23.” These are meaningful phrases, but we don’t have to suppose that their meaning consists in what they denote, as Russell did. They don’t denote at all. We can, however, deny their existence in the manner of (18.10). And, if we use the $\mathbf{E}$ predicate, we can even make it look like a subject/predicate proposition, as in (18.12). The strongest type of designation is a term that designates an existent. In this temporal world, Boris Johnson exists: the term “Boris Johnson” is meaningful, and it designates an existent. The middle case is a term that designates something in another possible world but not this one, like “George Washington” or “Lord Wellington.” Although they don’t exist, i.e., nothing *is* identical with them, they *did* exist. Finally, we

consider expressions like “the present king of France.” This designates something in an earlier temporal world, e.g., Louis XIV. We can say not that Louis XIV is the present King of France, but rather, that it was the case that Louis XIV is the present King of France. The expression “the present King of France” fails to designate in this temporal world, but it does in an earlier one. “The present King of France” is a nonrigid designator. So, to deny the existence of the present King of France, we must understand the denial as:

$$\neg\langle\lambda x.E(x)\rangle(\text{the present King of France}) \text{ or} \quad (18.14)$$

$$\neg E(\text{the present King of France}), \quad (18.15)$$

not as

$$\langle\lambda x.\neg E(x)\rangle(\text{the present King of France}) \text{ or} \quad (18.16)$$

$$\bar{E}(\text{the present King of France}). \quad (18.17)$$

George Washington, even though he fails to exist, does have properties. George Washington, for example, has the property of nonexistence: *He* does not exist. On the other hand, he has the property of *having existed*. Recall our use of **P** and **F** to abbreviate *It was the case that* and *It will be the case that*, respectively. Then, the following are both true:

$$\langle\lambda x.\neg E(x)\rangle(g) \quad (18.18)$$

$$\langle\lambda x.PE(x)\rangle(g). \quad (18.19)$$

We are saying *of* George Washington that he does not exist but he did. In each case, we are ascribing properties to him: the property of nonexistence, the property of past existence. And since “George Washington” is a rigid designator, *g* is invisible to the scope distinction, i.e., the following is also true:

$$P\langle\lambda x.E(x)\rangle(g). \quad (18.20)$$

George Washington has various positive properties, for example, the property of *having been* the president of the United States and the property of *having had* wooden teeth. It has sometimes been argued that nonexistent objects can have no positive properties because there is nothing to hang them on. They can, so the view has it, possess only negative properties. There seems to be a confusion in this reasoning that is nicely brought out by predicate abstraction notation. There are two places one can bring in negation: (a) one can ascribe happiness or *unhappiness*, existence or *nonexistence* to an individual; (b) one can ascribe or *not* ascribe happiness, existence, etc. to an individual. It is nigh impossible to distinguish positive and negative properties, as any number of philosophers and logicians have pointed out. But positive and negative *ascriptions* of properties can be distinguished

on purely syntactic grounds. For while (18.18) is an instance of the form $F(g)$ the following,

$$\neg\langle\lambda x.E(x)\rangle(g), \quad (18.21)$$

is not. We might say, then, *that* nonexistent objects have positive or negative properties as it seems reasonable; but for a term t that does not designate, there are no true sentences that say *of* t that it has a positive property, or a negative property.

How far can we ascribe properties to nonexistents? Is George Washington a President? Does George Washington have wooden teeth? These are all questions that demand an answer, in the sense that in the indicative forms, each of these has a truth value *in this world*. What is the truth value? It is not certain how we are to answer in each case. If we think that these predicates are true only of things in the domain of this world, then we will take each to be false. But we don't have to think this way. It is possible to take a more relaxed attitude toward these predicates and suppose that some will be true of George Washington and others false. It seems to us reasonable, for example, to say that George Washington is a President, though not the current one, for it seems right to suppose that anyone who was a President is a President. It also seems reasonable to suppose that George Washington is a man, even though he does not exist. It is not so clear what is to be said about his having wooden teeth. Essentially, these are not issues of what is true in a formal model, but rather, which formal model best reflects the way we use natural language. Reasonable people can debate this, but within each formal model, truth and falsity for sentences is clearly ascribed.

Exercises

Exercise 18.2.1 Let t be a term that does not designate. Are there any true sentences that say of t that it has a negative property?

Reference

Russell, B. (1905). On denoting. *Mind*, 14, 479–493. Reprinted in Robert C. Marsh, ed., *Logic and Knowledge: Essays 1901–1950*, by Bertrand Russell, Allen & Unwin, London, 1956.

Chapter 19

Rigidity



Kripke (1980) introduced the terminology of rigid designation into the modern literature on modal logic, though the notion was anticipated by Marcus (1992). In Kripke's terminology a *rigid designator* designates the same object in all possible worlds in which that object exists and never designates anything else. Kripke is usually thought to be requiring that the object must exist in the world in which it is designated, and he was taken to task by Kaplan (1989) who thought a term should be considered rigid if the object designated exists in any possible world. Our view follows Kaplan's: we say a singular term is a rigid designator if it refers to the same object in every possible world in which the term designates. It is worth noting that there are a number of notions of rigidity that have made their way into the philosophical literature. Kripke spoke of a term as *strongly rigid* if it designates a necessary existent, *de jure rigid* if it is stipulated to designate rigidly, and *de facto rigid* if it just happens to designate rigidly. In addition, we find the notion of a term that is *persistently rigid* if it designates the same object in every possible world in which it exists and does not designate anything different in worlds in which the object does not exist; and the notion of a term that is *obstinately rigid* if it designates the same object in every possible world, whether it exists in that world or not (Salmon, 2022).

In this chapter we examine a number of technical and philosophical issues concerning rigidity.

19.1 Rigid Designators

Rigidity can be characterized in a straightforward way, as we saw above. In Proposition 19.2.6 we will see that rigidity of a singular term c is closely connected with the validity of all instances of the following:

$$\langle \lambda x. \Box \Phi(x) \rangle(c) \equiv \Box \langle \lambda x. \Phi(x) \rangle(c).$$

That is, a rigid designator is a designator for which scope does not matter. It should be clear that although we can define rigidity in terms of scope, the notion of scope is much broader: as Kripke (1979) has argued, none of the family of binary distinctions—rigid/nonrigid, *de re/de dicto*, referential/attributional—can replace the scope distinction, any more than the primary/secondary distinction of Russell (1905) can. After all, we can have nesting of modal operators, say, or multi-variable predicates, and the richness of the interpretations cannot be captured by a simple on/off feature analysis.

But the notion so defined speaks about terms in a *formal language*. The question remains whether any *natural language* expressions should be represented by rigid designators when formalized. It should be noted, incidentally, it is possible that a term might be rigid under one interpretation of $\Box$ and $\Diamond$ but not under another. So, when we speak about the rigidity of natural language expressions, we have to specify the interpretation of the modal operators.

Let us, for now, suppose that we are working with the usual alethic modalities. Then it turns out that some definite descriptions are rigid, but not all. “The number of U.S. Supreme Court Justices” is non-rigid. Although there are, as a matter of fact, 9 Supreme Court Justices, there is no reason why, in other possible worlds, there couldn’t be fewer justices, or more. And the nonrigidity of the expression shows itself in the fact that the two following sentences differ in truth value:

The number of Supreme Court Justices is necessarily odd (19.1)

and

Necessarily, the number of Supreme Court Justices is odd. (19.2)

The *de re* reading (19.1) is true because there are 9 Supreme Court Justices, and this number is, in every possible world, odd. The *de dicto* reading (19.2) on the other hand, is false, because the number of Supreme Court Justices can vary from world to world, and in some, that number could be even.

On the other hand, “the ratio of the circumference to the diameter of a circle” is rigid, for it designates the same number, π , in every possible world. In this case, the two sentences

The ratio of the circumference of a circle to its diameter
is necessarily irrational

and

Necessarily, the ratio of the circumference of a circle to
its diameter is irrational.

have the same truth value.

In the case of proper names, the situation is a bit more complicated. Kripke (1980) has argued that ordinary proper names are rigid when the modalities are understood alethically. The reference of a proper name is fixed by some arbitrary procedure in which names are assigned to individuals—a baptismal ceremony, perhaps. But once the reference has been fixed, that name refers to the individual it has been assigned to, in every possible world. When we consider whether it is possible (say) for Cicero to be a modal logician, we consider whether there is a possible world in which *that person* who had been baptized “Cicero” is a modal logician.

Kripke’s *causal/historical* account of proper names is to be contrasted with the *description theory* associated with Russell (1905) and Frege (1892). Russell and Frege regarded a proper name as short for some definite description. “Homer,” for example, means “the author of the *Illiad*.” And if these two mean the same thing, then in every possible world, they will refer to the same object. But this, Kripke has pointed out, does not seem right, because it is only a contingent property of Homer that he wrote the *Illiad*. On the description theory, however, Homer will be whoever writes the *Illiad* in that world, and since this can vary, the reference of “Homer” will vary from world to world. By contrast, Kripke’s view is that the reference of a name is fixed, not by an associated description, but originally, by a “baptism,” and subsequently, by the intention of the speakers of the language to maintain the reference of the original baptism.

On Kripke’s view, then, where $\Box$ and $\Diamond$ are interpreted alethically, if a and b are proper names, they are rigid. From this the equivalence of $\langle \lambda x, y. \Box(x = y) \rangle(a, b)$ and $\Box \langle \lambda x, y. x = y \rangle(a, b)$ follows, and so we might write $\Box(a = b)$ as shorthand for both. So in particular, from the truth of (19.3),

$$\text{Hesperus} = \text{Phosphorus}, \quad (19.3)$$

(19.4) follows:

$$\Box(\text{Hesperus} = \text{Phosphorus}). \quad (19.4)$$

This, however, reawakens the old puzzle about the morning star and the evening star and what the ancients really knew (and when). If we take *a* to be *Hesperus* and *b* to be *Phosphorus*, and $\Box$ to be *The ancients knew that*, we have the unfortunate result that, if Hesperus is Phosphorus, then the ancients knew that this was so. And, it appears, they did not. The problem, quite clearly, lies with the understanding that these names are rigid designators. We see that, although they are rigid within the context of an *alethic* reading of $\Box$, they cannot be rigid under an *epistemic* reading of $\Box$. In fact, it is hard to see whether there are any rigid designators under the epistemic reading. Kripke distinguished very clearly between what is metaphysically necessary but, from an epistemological perspective is *a posteriori*. Such is the case for (19.3). But he also held that there are statements that are *a priori* but metaphysically contingent.

Finally, we mention one nice outcome of our way of dealing with rigidity, namely, we can extend the notion of rigidity to general terms in a natural way. Kripke's original definition does not extend readily to general terms. With his notion of rigidity we cannot say, for example, that "tiger" designates the same objects in every possible world because the extension of "tiger" in one world need not be the same as the extension of "tiger" in another if, as we suppose, we are dealing with a variable domain model. But once rigidity and existence have been separated, we can say that a natural kind term $F(x)$ is rigid if it is an essential property of anything that possesses it, i.e.,

$$\langle \lambda x. \Diamond F(x) \rangle(y) \equiv \langle \lambda x. \Box F(x) \rangle(y). \quad (19.5)$$

Our use of free variables enables us to speak generally over the domain of the frame.

But this is only one way of extending the notion of rigidity to general terms. Kripke actually thought a general term like "tiger" designated a *kind* of object, and the term designated the same kind in every possible world. For Kripke's handling of these cases, see Soames (2002).

Exercises

Exercise 19.1.1 Are the following two claims equivalent?

1. A rigid designator is one for which scope does not matter.
2. A rigid designator is one which is always given wide scope.

19.2 Rigidity Formally

Since we depart somewhat from Kripke, here is a proper statement of the official terminology used in the present book.

Definition 19.2.1 (Rigidity) Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a model.

1. A constant symbol c is *rigid* in $\mathcal{M}$ if c designates the same thing in each world of the model at which it designates. More precisely, c is rigid if $\mathcal{I}(c, \Gamma)$ is the same for all $\Gamma \in \mathcal{G}$ for which $\mathcal{I}(c, \Gamma)$ is defined. Similarly for function symbols.
2. A constant symbol c is *strongly rigid* in $\mathcal{M}$ if c is rigid and always designates. That is, c is strongly rigid if $\mathcal{I}(c, \Gamma)$ is defined for all $\Gamma \in \mathcal{G}$, and has the same value for all Γ . Again, similarly for function symbols.

We have both rigidity and strong rigidity to consider, and similarly for a local version as defined below. Strong rigidity is technically much the simpler of the two. In the rest of this section we examine some nice properties that it has, and turn to rigidity without the ‘strong’ qualifier in Sect. 19.3.

Strong rigidity is a global notion—something is the case at every possible world of a model. It turns out that there is also a very simple and useful local version.

Definition 19.2.2 (Local Strong Rigidity) Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a model. We say a constant symbol c is *locally strongly rigid* in $\mathcal{M}$ if c always designates and, for any $\Gamma, \Delta \in \mathcal{G}$ such that $\Gamma \mathcal{R} \Delta$, we have $\mathcal{I}(c, \Gamma) = \mathcal{I}(c, \Delta)$. Similarly for function symbols.

Thus rigidity refers to *all* worlds, while local rigidity only refers to *related* worlds. Rigidity and local rigidity are different things; it is easy to produce examples where some constant symbol is locally strongly rigid, but is not strongly rigid. Nonetheless, the two versions characterize the same logics, in the following sense.

Proposition 19.2.3 *Let S be a set of constant and function symbols. The following are equivalent for any formula Φ :*

1. Φ is valid in all models in which the members of S are strongly rigid;
2. Φ is valid in all models in which the members of S are locally strongly rigid.

Before giving the proof we introduce some terminology.

Definition 19.2.4 (Relevant World) Suppose we have a model with Γ, Δ as possible worlds. And suppose there is a path $\Omega_1, \Omega_2, \dots, \Omega_n$ in the model that takes us from Γ to Δ . That is, we have $\Gamma = \Omega_1, \Omega_1 \mathcal{R} \Omega_2, \Omega_2 \mathcal{R} \Omega_3, \dots, \Omega_{n-1} \mathcal{R} \Omega_n$, and $\Omega_n = \Delta$, where $\mathcal{R}$ is the accessibility relation of the model. Then we say Δ is *relevant* to Γ .

The path length n can be 1, so we have $\Gamma = \Omega_1 = \Gamma$ telling us worlds are relevant to themselves. The path length can be 2, so we have $\Gamma = \Omega_1 \mathcal{R} \Omega_2 = \Delta$, telling us that any world accessible from Γ is relevant to Γ . Similarly any world accessible from a world accessible from Γ is relevant to Γ , and so on. The reason this is important is very simple. If we are trying to determine whether or not a particular formula holds at Γ in the model, the only worlds we need to take into consideration are worlds that are relevant to Γ . We now turn to the proof of the Proposition above.

Proof We first show item 2 implies item 1 and the argument, for the contrapositive form, is very simple. Suppose 1 fails, and Φ is not valid in some model in which the

members of S are strongly rigid. Trivially the members of S are also locally strongly rigid, so Φ is not valid in some model in which the members of S are locally strongly rigid, and thus 2 fails.

Showing item 1 implies item 2 is also done contrapositively. Suppose 2 fails, and Φ is not valid in some model in which the members of S are locally strongly rigid. We produce another model in which Φ is not valid, and in which the members of S are (simply) strongly rigid.

Since Φ is not valid in the model, there is some world Γ at which Φ fails (with respect to some valuation). Construct a new model in which the set of possible worlds consists of all those worlds that are relevant to Γ (which, in particular, includes Γ). Accessibility in the new model is the original accessibility relation restricted to the new domain, the domain function is the original one restricted to the new domain, and similarly for the interpretation function. Thus we have fully specified our new model.¹

Every possible world Δ that is relevant to Γ in the original model must interpret each member of S the same way that Γ does, since the interpretation is preserved by each accessibility relation instance in any chain connecting Γ to Δ , because we have local strong rigidity for members of S . Thus the new model is, simply, strongly rigid for members of S . And a proof using Complete Induction on formula complexity will verify that formulas have the same truth values at Γ in both the original model and in the new one. (Indeed, this happens at every world of the new model.) Thus Φ fails at Γ in the new model, in which the members of S are strongly rigid. Hence item 1 fails. $\square$

Now that we know strong rigidity and local strong rigidity are equivalent as far as formula validity goes, we can work with whichever is convenient, and for the rest of this section that is *local* strong rigidity.

We are about to show that strong rigidity, which is a semantic concept, corresponds via local strong rigidity to a syntactic notion. In fact this, syntactic notion can be given in several versions whose equivalence is not at all obvious. We present the results for a constant symbol c to keep the formulas simple, though similar results apply to function symbols and c can be replaced with $f(x_1, \dots, x_n)$ throughout the following. We have seen many examples where syntactic properties corresponded to semantic properties, but these have been all been semantic properties of *frames* and not of particular models. We are about to see yet another such example.

Definition 19.2.5 ($\mathcal{I}_0$ -compatible) Let c be a constant symbol. Let $\mathcal{F} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ be a skeleton (Definitions 8.6.1 and 8.7.1) and let $\mathcal{I}_0$ be what we call a c -interpretation in it, defined on c and on nothing else. That is, for each $\Gamma \in \mathcal{G}$, $\mathcal{I}_0(c, \Gamma)$ is defined, but $\mathcal{I}_0$ is undefined on other constant symbols and on all function and relation symbols.

¹ In the literature the new model we constructed is called the submodel of the original model generated by $\{\Gamma\}$. Any subset of the original set of worlds can be used to generate a submodel in a similar way.

We say $\mathcal{I}_0$ interprets c *locally strongly rigidly* provided, for any $\Gamma, \Delta \in \mathcal{G}$, if $\Gamma \mathcal{R} \Delta$, then $\mathcal{I}_0(c, \Gamma) = \mathcal{I}_0(c, \Delta)$.

We say a model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ that is based on the skeleton $\mathcal{F}$ is $\mathcal{I}_0$ -*compatible* provided $\mathcal{I}$ and $\mathcal{I}_0$ agree on c at each world of $\mathcal{G}$.

The idea is, if $\mathcal{I}_0$ interprets c locally strongly rigidly, then when we talk of $\mathcal{I}_0$ -compatible models based on $\mathcal{F}$ we are really talking about members of a family of models in which the interpretation is fixed on c and is locally strongly rigid, though interpretations of other items can vary.

Proposition 19.2.6 *Let $\mathcal{F} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D} \rangle$ be a skeleton, and $\mathcal{I}_0$ be a c -interpretation in this skeleton. The following are equivalent:*

1. $\mathcal{I}_0$ interprets c locally strongly rigidly.
2. All instances of the schema $\Box \langle \lambda x. \Phi \rangle (c) \equiv \langle \lambda x. \Box \Phi \rangle (c)$ are valid in every $\mathcal{I}_0$ -compatible model based on $\mathcal{F}$.
3. All instances of the schema $\Box \langle \lambda x. \Phi \rangle (c) \supset \langle \lambda x. \Box \Phi \rangle (c)$ are valid in every $\mathcal{I}_0$ -compatible model based on $\mathcal{F}$.
4. All instances of the schema $\langle \lambda x. \Box \Phi \rangle (c) \supset \Box \langle \lambda x. \Phi \rangle (c)$ are valid in every $\mathcal{I}_0$ -compatible model based on $\mathcal{F}$.

We give the proof below, but first we have some important observations. Item 2 says that for the constant c , *de re* and *de dicto* distinctions cannot be seen. The equivalence between 1 and 2 essentially says that the lack of *de re* and *de dicto* distinctions is characteristic of local strong rigidity. What is somewhat unexpected is the further equivalence between this and items 3 and 4 separately. These latter say that either half of the equivalence in item 2 suffices. Exercise 19.2.3 gives further equivalent *de re/de dicto* schemas. Now we turn to the proof.

Proof Showing that item 1 implies the other three is rather easy, and we leave it as an exercise.

Now we show that if item 1 is false, so is item 3, and hence also item 2, so either of 2 or 3 implies 1. So, suppose $\mathcal{I}_0$ does not interpret c locally strongly rigidly. Say there are worlds, $\Gamma, \Delta \in \mathcal{G}$, with $\Gamma \mathcal{R} \Delta$, but $\mathcal{I}_0(c, \Gamma) \neq \mathcal{I}_0(c, \Delta)$. We use this information to falsify item 3.

We define a particular $\mathcal{I}_0$ -compatible model by defining a suitable interpretation $\mathcal{I}$ as follows. On c , $\mathcal{I}$ and $\mathcal{I}_0$ agree, and on other function and constant symbols, make some arbitrary choice (it won't matter). Thus $\mathcal{I}$ and $\mathcal{I}_0$ agree on c , so the model we are constructing will be $\mathcal{I}_0$ -compatible. Let P be a one-place relation symbol. For each world $\Omega \in \mathcal{G}$, set $\mathcal{I}(P, \Omega) = \{\mathcal{I}_0(c, \Omega)\}$. That is, at each world we take P to be true of exactly what c designates at that world—informally, we can read P as “is c .” On other relation symbols, again make some arbitrary choice (it, too, won't matter). We thus have our $\mathcal{I}_0$ -compatible model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$.

We first check that we have $\mathcal{M}, \Gamma \Vdash_v \Box \langle \lambda x. P(x) \rangle (c)$ (for any v). This will be the case provided that for any $\Omega \in \mathcal{G}$ such that $\Gamma \mathcal{R} \Omega$ we have $\mathcal{M}, \Omega \Vdash_v \langle \lambda x. P(x) \rangle (c)$, and this is so because, by definition, $\mathcal{I}(c, \Omega) \in \mathcal{I}(P, \Omega)$.

We next check that we do *not* have $\mathcal{M}, \Gamma \Vdash_v \langle \lambda x. \Box P(x) \rangle (c)$. Well if we did, we would have $\mathcal{M}, \Gamma \Vdash_{v'} \Box P(x)$, where v' is the x -variant of v such that $v'(x) = \mathcal{I}(c, \Gamma)$. And if we had this, since $\Gamma \mathcal{R} \Delta$, we would have $\mathcal{M}, \Delta \Vdash_{v'} P(x)$, and thus $v'(x) \in \mathcal{I}(P, \Delta)$. However by definition, the only member of $\mathcal{I}(P, \Delta)$ is $\mathcal{I}_0(c, \Delta)$, but $v'(x) = \mathcal{I}_0(c, \Gamma)$, and $\mathcal{I}_0(c, \Gamma) \neq \mathcal{I}_0(c, \Delta)$.

It follows that we have

$$\mathcal{M}, \Gamma \not\Vdash_v \Box \langle \lambda x. P(x) \rangle (c) \supset \langle \lambda x. \Box P(x) \rangle (c)$$

And thus the failure of item 1 implies the failure of item 3. We leave item 4 to you as an exercise. $\square$

The proof given for Proposition 19.2.6 is semantic, but in fact there is constructive content available. Each instance of schema 3 is implied by an instance of schema 4, and each instance of schema 4 is implied by an instance of schema 3, and we can say exactly what instances are needed.

Let $\Phi(x)$ be a formula, and assume the variable y does not appear in it. Let $\Phi(y)$ be the result of substituting y for free occurrences of x in $\Phi(x)$. We now define four formulas, all involving the constant symbol c .

$$\begin{aligned} A_3 &= \Box \langle \lambda y. \langle \lambda x. \Phi(y) \supset \Phi(x) \rangle (c) \rangle (c) \supset \\ &\quad \langle \lambda y. \Box \langle \lambda x. \Phi(y) \supset \Phi(x) \rangle (c) \rangle (c) \\ A_4 &= \langle \lambda x. \Box \Phi(x) \rangle (c) \supset \Box \langle \lambda x. \Phi(x) \rangle (c) \\ B_4 &= \langle \lambda y. \langle \lambda x. \Box (\Phi(x) \supset \Phi(y)) \rangle (c) \rangle (c) \supset \\ &\quad \langle \lambda y. \Box \langle \lambda x. \Phi(x) \supset \Phi(y) \rangle (c) \rangle (c) \\ B_3 &= \Box \langle \lambda x. \Phi(x) \rangle (c) \supset \langle \lambda x. \Box \Phi(x) \rangle (c) \end{aligned}$$

Notice that A_4 and B_3 are arbitrary cases of items 4 and 3 of Proposition 19.2.6, while A_3 and B_4 are actually particular instances of items 3 and 4 respectively. Now it can be shown that $A_3 \supset A_4$ and $B_4 \supset B_3$ are both valid (in varying domain $\mathbf{K}$, say). Showing this is a good work-out for tableaux, and we give it as Exercise 19.2.5.

Recall that both the Barcan and the Converse Barcan schemas could be replaced with single formulas that use equality, Sect. 12.8. A similar thing happens here: local strong rigidity can be captured by a single formula (assuming we work with *normal* models).

Proposition 19.2.7 *The constant symbol c is locally strongly rigid in a normal model if and only $\langle \lambda y. \Box \langle \lambda x. x = y \rangle (c) \rangle (c)$ is valid in that model.*

Proof Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a normal model. Just for this proof, let us say constant symbol c is locally strongly rigid at possible world $\Gamma \in \mathcal{G}$ if, for every $\Delta \in \mathcal{G}$ with $\Gamma \mathcal{R} \Delta$, $\mathcal{I}(c, \Gamma) = \mathcal{I}(c, \Delta)$. It is easy to check that for

any $\Gamma \in \mathcal{G}$, c is locally strongly rigid at Γ if and only if for each valuation v , $\mathcal{M}, \Gamma \Vdash_v \langle \lambda y. \Box \langle \lambda x. x = y \rangle (c) \rangle (c)$. Then c is locally strongly rigid in $\mathcal{M}$ if and only if c is locally strongly rigid at each possible world Γ if and only if $\mathcal{M}, \Gamma \Vdash_v \langle \lambda y. \Box \langle \lambda x. x = y \rangle (c) \rangle (c)$ is true at each possible world Γ for every v if and only if $\langle \lambda y. \Box \langle \lambda x. x = y \rangle (c) \rangle (c)$ is valid in the model. $\square$

Exercises

Exercise 19.2.1 Show that item 1 of Proposition 19.2.6 implies item 2, and hence trivially items 3 and 4.

Exercise 19.2.2 Show that if item 1 of Proposition 19.2.6 is false, so is item 4.

Exercise 19.2.3 Extend Proposition 19.2.6 by showing items 1–4 are further equivalent to

5. All instances of the schema $\Diamond \langle \lambda x. \Phi \rangle (c) \equiv \langle \lambda x. \Diamond \Phi \rangle (c)$ are valid in every $\mathcal{I}_0$ -compatible model based on $\mathcal{F}$.
6. All instances of the schema $\Diamond \langle \lambda x. \Phi \rangle (c) \supset \langle \lambda x. \Diamond \Phi \rangle (c)$ are valid in every $\mathcal{I}_0$ -compatible model based on $\mathcal{F}$.
7. All instances of the schema $\langle \lambda x. \Diamond \Phi \rangle (c) \supset \Diamond \langle \lambda x. \Phi \rangle (c)$ are valid in every $\mathcal{I}_0$ -compatible model based on $\mathcal{F}$.

Exercise 19.2.4 Definition 19.2.5 and Proposition 19.2.6 are formulated for constant symbols. Now provide a version of each appropriate for non-rigid function symbols, and see what modifications of the proof of Proposition 19.2.6 are needed to make the result go through with function symbols.

Exercise 19.2.5 This exercise fills in missing work in this section. Recall, we defined four formulas as follows.

$$\begin{aligned}
 A_3 &= \Box \langle \lambda y. \langle \lambda x. \Phi(y) \supset \Phi(x) \rangle (c) \rangle (c) \supset \\
 &\quad \langle \lambda y. \Box \langle \lambda x. \Phi(y) \supset \Phi(x) \rangle (c) \rangle (c) \\
 A_4 &= \langle \lambda x. \Box \Phi(x) \rangle (c) \supset \Box \langle \lambda x. \Phi(x) \rangle (c) \\
 B_4 &= \langle \lambda y. \langle \lambda x. \Box (\Phi(x) \supset \Phi(y)) \rangle (c) \rangle (c) \supset \\
 &\quad \langle \lambda y. \Box \langle \lambda x. \Phi(x) \supset \Phi(y) \rangle (c) \rangle (c) \\
 B_3 &= \Box \langle \lambda x. \Phi(x) \rangle (c) \supset \langle \lambda x. \Box \Phi(x) \rangle (c)
 \end{aligned}$$

Give tableau proofs, using varying domain **K** rules assuming terms always designate, that is, VA conditions, of the following.

1. $A_3 \supset A_4$.
2. $B_4 \supset B_3$.

Note: the choice between varying and constant domains is not actually relevant since there are no quantifiers. But the proof does not go through if terms might not designate.

Exercise 19.2.6 Prove each of the following using the VN version of the **K** rules.

1. $\langle \lambda y. \Box \langle \lambda x. x = y \rangle (c) \rangle (c) \supset [\Box \langle \lambda x. P(x) \rangle (c) \supset \langle \lambda x. \Box P(x) \rangle (c)]$
2. $\langle \lambda y. \Box \langle \lambda x. x = y \rangle (c) \rangle (c) \supset [\langle \lambda x. \Box P(x) \rangle (c) \supset \Box \langle \lambda x. P(x) \rangle (c)]$

19.3 Rigidity That Isn't Strong

In the previous section *strong* rigidity was the central topic. Being able to take designation for granted greatly simplifies things. The technical work extends to rigidity that is not strong, but the details are somewhat complicated. Feel free to skip this section.

A key ingredient that went into the results about strong rigidity was the notion of *local strong* rigidity. The object that is designated at a world is also the one designated at related worlds. The replacement for this now, which we call just *local rigidity*, is a kind of action at a distance.

Definition 19.3.1 (Local Rigidity) Let $\mathcal{M}$ be a model. We say a constant symbol c is *locally rigid* in $\mathcal{M}$ provided, if c designates at both the possible worlds Γ and Δ , and Δ is relevant to Γ , Definition 19.2.4, then c designates the same object at Γ and Δ . Similarly for function symbols.

A central result earlier was Proposition 19.2.3, but the proof of it does not go through if strongness is not assumed. Still, the general outline can be saved. We begin by introducing a specialized notion of validity. The idea is, in a model in which constant and function symbols are rigid, but not strongly so, we concentrate on the worlds at which these symbols do designate.

Definition 19.3.2 (Designation Validity) Let S be a set of constant and function symbols. We say a formula Φ is *S designation valid* in a model, with respect to a valuation, if it is true at all possible worlds of the model at which the members of S actually designate.

Now, here is our analog of Proposition 19.2.3, allowing us to replace rigidity with a local version.

Proposition 19.3.3 *Let S be a set of constant and function symbols. The following are equivalent for any formula Φ :*

1. Φ is *S designation valid* in all models in which the members of S are rigid.
2. Φ is *S designation valid* in all models in which the members of S are locally rigid.

Proof Once again we prove that the negation of each item implies the negation of the other. And again, half is simple. Suppose 1 fails, and Φ is not S designation valid in some model in which the members of S are rigid. A rigid model is also a locally rigid model, so Φ is not S designation valid in some model in which the members of S are locally rigid, and so 2 fails.

Now suppose 2 fails. Then we have a model in which the members of S are locally rigid, and a possible world Γ at which all the members of S designate, but at which Φ is not true (with respect to some valuation).

As before, form a new model in which the set of possible worlds consists of Γ and all those worlds that are relevant to Γ . And, just as before, accessibility in the new model is the original accessibility relation restricted to the new domain, the domain function is the original one restricted to the new domain, and similarly for the interpretation function. We now have fully specified a submodel of our original one.

As before, a Complete Induction shows that formula truth or falsity is the same at worlds common to the two models. All that is left is to verify that the new submodel interprets the constant and function symbols of S locally rigidly, and this is easy. If Δ is a world in the new model, it must be relevant to Γ . If, also, Δ is a world at which the members of S designate, then since they also designate at Γ they must designate the same values as at both Γ and Δ since we have local rigidity. It follows that we have a model that interprets the members of S rigidly. Thus we have a model showing that item 1 fails. $\square$

We skip over any discussion of an analog to Proposition 19.2.6, and move to a somewhat informal discussion of one for Proposition 19.2.7. The formula $\langle \lambda y. \Box \langle \lambda x. x = y \rangle (c) \rangle (c)$ from that proposition was used in a context having a strongness assumption. Without strongness, we could try modifying the formula to make designation assumptions explicit, as follows.

$$\mathbf{D}(c) \supset \langle \lambda y. \Box (\mathbf{D}(c) \supset \langle \lambda x. x = y \rangle (c)) \rangle (c)$$

Without having a strongness assumption, validity of this formula captures the idea that, if c happens to designate at two consecutive worlds of one of our paths starting at Γ , Definition 19.2.4, then c will designate the same thing at both worlds. But what if worlds are not consecutive in a path, and we have $\Gamma \mathcal{R} \Delta \mathcal{R} \Omega$, where c might not designate in the middle but does so at both ends. Well, consider the following formula.

$$\mathbf{D}(c) \supset \langle \lambda y. \Box \Box (\mathbf{D}(c) \supset \langle \lambda x. x = y \rangle (c)) \rangle (c)$$

Validity of this tells us that if c designates at possible world Γ , and $\Gamma \mathcal{R} \Delta \mathcal{R} \Omega$ and c designates again at Ω , then c will designate the same thing at both Γ and Ω , even though there is a world in between. Likewise

$$\mathbf{D}(c) \supset \langle \lambda y. \Box^3 (\mathbf{D}(c) \supset \langle \lambda x. x = y \rangle (c)) \rangle (c)$$

allows skipping over two intermediate worlds. And so on.

Proposition 19.2.7 tells us that identical designation at related worlds is equivalent to the validity of $\langle \lambda y. \Box \langle \lambda x. x = y \rangle (c) \rangle (c)$. We have just seen how we could capture identical designation at worlds two moves apart, or three moves apart, and so on. Putting all this together carefully, we have the provability of the following.

Proposition 19.3.4 *The constant symbol c is locally rigid in a normal model if and only the following are all $\{c\}$ designation valid, Definition 19.3.2, in that model:*

$$\mathbf{D}(c) \supset \langle \lambda y. \Box^n (\mathbf{D}(c) \supset \langle \lambda x. x = y \rangle (c)) \rangle (c) \quad n = 1, 2, 3, \dots$$

We do not carry this investigation out further here, since it would be rather specialized. We leave it to the sufficiently interested reader. We do note one significant simplification that can be achieved, though. The discussion above made no specific assumptions about the underlying modal logic involved—in effect, it is **K**. But if the logic is as strong as **K4**, the formula set above can be replaced by the following single formula

$$\mathbf{D}(c) \supset \langle \lambda y. \Box (\mathbf{D}(c) \supset \langle \lambda x. x = y \rangle (c)) \rangle (c)$$

because in **K4**, $\Box X \supset \Box \Box X \supset \Box \Box \Box X \dots$

19.4 A Dynamic Logic Example

At the end of Sect. 4.4 we briefly mentioned *dynamic logic* (Pratt, 1976; Harel, 1984; Harel et al., 2012), and we now have more to say about it. Dynamic logic is a version of modal logic in which there are many modal operators, each corresponding to an action—typically, the action of executing a computer program. Since the variables used in most computer languages change their values from time to time, dynamic logic provides us with an example of non-rigid constant symbols that is natural, but of a different kind than we have been considering.

Suppose c is a variable in the sense of computer programming, and which we will think of here as a non-rigid constant symbol in a modal language. Consider the assignment statement $c := c + 1$, which increments the value of c . Corresponding to this is the dynamic logic modal operator $\boxed{c := c+1}$, where the intention is that $\boxed{c := c+1} \Phi$ should mean: after the assignment statement $c := c + 1$ is executed, Φ will be true. For notational convenience, in what follows we will abbreviate $\boxed{c := c+1}$ as $\Box$.

To express the intended behavior of $\Box$ one might be tempted to write something like the following, where we assume arithmetic notation is interpreted in the usual way.

$$(\forall x)[(c = x) \supset \Box(c = x + 1)] \quad (19.6)$$

Informally, if the current value of c is x , after the assignment statement $c := c + 1$ has been executed, the value of c will be $x + 1$. As written, the formulation makes no use of predicate abstraction, and this omission immediately leads us to a serious problem. In classical first-order logic the following is valid:

$$(\forall x)[(c = x) \supset P(x)] \equiv P(c). \quad (19.7)$$

Using this with $P(x)$ being $\Box(c = x + 1)$, (19.6) immediately yields:

$$\Box(c = c + 1) \quad (19.8)$$

which is silly. Indeed, silliness came in at the very beginning. For instance, we wrote $c = x$ in (19.7), but c is a non-rigid constant symbol while x is an object, namely an integer. It makes no sense to equate these different kinds of things. Clearly some use of predicate abstraction is essential.

Using predicate abstraction, (19.6) can be restated correctly as follows:

$$(\forall x)[\langle \lambda y. y = x \rangle(c) \supset \Box \langle \lambda y. y = x + 1 \rangle(c)]. \quad (19.9)$$

Then $\langle \lambda y. y = x \rangle(c)$ is true at a particular possible world in a model just in case c designates the number x , and $\langle \lambda y. y = x + 1 \rangle(c)$ is true just in case c designates the number $x + 1$.

In modal logic using predicate abstraction, we have the validity of the following (assuming c always designates), which is similar to the incorrect (19.7).

$$(\forall x)[\langle \lambda y. y = x \rangle(c) \supset P(x)] \equiv \langle \lambda x. P(x) \rangle(c). \quad (19.10)$$

Using (19.10) with $P(x)$ being $\Box \langle \lambda y. y = x + 1 \rangle(c)$, the correct (19.9) can be replaced by the following.

$$\langle \lambda x. \Box \langle \lambda y. y = x + 1 \rangle(c) \rangle(c) \quad (19.11)$$

This formula should be compared with the incorrect (19.8).

This example can be explored a little further yet. The three sentences

1. $\mathbf{E}(c)$ (Definition 15.4.3)
2. $(\forall x)\Box\neg(x = x + 1)$
3. $\langle \lambda x. \Box \langle \lambda y. y = x + 1 \rangle(c) \rangle(c)$

together imply

$$4. \neg \langle \lambda x. \Box \langle \lambda y. x = y \rangle (c) \rangle (c)$$

in the logic **D**. (We gave this as Exercise 15.5.9, using tableaux.) Sentence 1 corresponds, in computer programming terminology, to c being initialized. Sentence 2 asserts that x and $x + 1$ are never the same, which is all we need from arithmetic now. Sentence 3 captures the behavior of the assignment statement $c := c + 1$ (recall, we are thinking of $\Box$ as being the action of executing this program step). Inference is in the logic **D**, not **K**. This amounts to assuming that $c := c + 1$ is executable—there must always be accessible states that $c := c + 1$ takes us to. Finally, the consequence, 4, expresses the (obvious) fact that c is not rigid, by Proposition 19.2.7. More generally, arithmetic variables in programming languages are always allowed to be non-rigid. It is, of course, how they usefully differ from constants of programming languages.

References

- Frege, G. (1892). Über Sinn und Bedeutung. *Zeitschrift für Philosophie und philosophische Kritik*, 100, 25–50. “On Sense and Reference” translated in Frege (1952).
- Frege, G. (1952). In P. Geach, & M. Black (Eds.), *Translations from the philosophical writings of Gottlob Frege*. Basil Blackwell.
- Gabbay, D. M. & Guenther, F. (Eds.). (1983–1989). *Handbook of philosophical logic* (Four volumes). Kluwer.
- Gabbay, D. M. & Guenther, F. (Eds.). (2001 to present). *Handbook of philosophical logic* (Multiple volumes, 2nd ed.). Springer/Kluwer.
- Harel, D. (1984). In D. M. Gabbay, & F. Guenther (Eds.), *Dynamic logic* (Chap. 10, Vol. 2, pp. 497–604). D. Reidel. 1983–1989
- Harel, D., Kozen, D., & Tiuryn, J. (2012). In D. M. Gabbay, & F. Guenther (Eds.), *Dynamic logic* (Vol. 4, pp. 99–217). D. Reidel. 2001 to present.
- Kaplan, D. (1989). Demonstratives. In J. Almog, J. Perry, & H. Wettstein (Eds.), *Themes from Kaplan* (pp. 481–564). Oxford University Press.
- Kripke, S. (1979). Speaker’s reference and semantic reference. In P. French, T. Uehling, & H. Wettstein (Eds.), *Contemporary perspectives in the philosophy of language* (pp. 6–27). University of Minnesota Press.
- Kripke, S. (1980). *Naming and necessity*. Harvard University Press. Reissued by Blackwell Publishing, 1991.
- Marcus, R. B. (1992). *Modalities*. Oxford University Press.
- Pratt, V. R. (1976). Semantical considerations on Floyd-Hoare logic. In *Proceedings of the 17th symposium on the foundations of computer science* (pp. 109–121). IEEE.
- Russell, B. (1905). On denoting. *Mind*, 14, 479–493. Reprinted in Robert C. Marsh, ed., *Logic and Knowledge: Essays 1901–1950*, by Bertrand Russell, Allen & Unwin, London, 1956.
- Salmon, N. (2022). Some notions of rigid designation. In *Handout to a talk at the “naming and necessity at 50 conference”*, Hudson, New York, July 19–22, 2022.
- Soames, S. (2002). *Beyond rigidity: the unfinished semantic agenda of naming and necessity*. Oxford University Press.

Chapter 20

Definite Descriptions



We have used phrases like “the King of France” or “the tallest person in the world” several times, though we always treated them like non-rigid constant symbols. But such phrases have more structure than constant symbols—they do not arbitrarily designate. The King of France, for instance, has the property of being King of France, provided there is one, and the phrase “the King of France” designates him because he alone has that property. Phrases of the form “the so-and-so” are called *definite descriptions*. In this chapter we examine the behavior of definite descriptions in modal contexts.

20.1 Two Theories of Descriptions

Two types of theories of descriptions are well-known in the literature: (a) those based on Frege’s treatment; (b) those based on Russell’s treatment. The account of descriptions presented in this text falls into neither of these two categories neatly, as we shall see in the next section.

Frege (1893) considered a definite description to be a genuine singular term (as do we), so that a sentence like

The present King of France is bald, (20.1)

would have the same logical form as “Harry Truman is bald.” What happens if there is no present King of France? To avoid truth value gaps (or the introduction of a third truth value), Frege arbitrarily assigned a default designatum to the description, namely, the expression itself, “the present King of France.” Carnap, working within Frege’s paradigm, took a somewhat narrower approach, having a single default designatum rather than a family of them. On his approach, the present King of France turns out to be identical with the present Queen of France, because France

has neither a King nor Queen, and so both terms designate the same default object. Apart from the arbitrariness of the default designatum, there remains something unsatisfying about the idea that the phrase “the King of France” designates, but not a King of France. One also wonders how denials of existence are to be treated.

Russell (1905), on the other hand, denied that definite descriptions were genuine singular terms. Descriptions were, on his view, “incomplete expressions,” and he provided a *contextual* account of descriptions. Sentence (20.1) is rephrased so that the apparent singular term is eliminated:

At least one thing presently kings France, and

At most one thing presently kings France, and

Whatever presently kings France is bald.

The first two conjuncts express existence and uniqueness conditions, respectively; the third conjunct expresses that the uniquely existent individual has a certain property. Let $K(x)$ abbreviate “ x presently kings France” and $B(x)$ abbreviate “ x is bald.” Then (20.1) can be symbolized as

$$(\exists x)((K(x) \wedge (\forall y)(K(y) \supset y = x)) \wedge B(x)). \quad (20.2)$$

But this paraphrase of sentences involving descriptions is only one part of Russell’s celebrated theory. We must not overlook the scope distinction, only dimly understood in Russell (1905), and presented there hurriedly as the much weaker distinction between a *primary* and *secondary* occurrence of a description. (A full treatment is presented in Whitehead and Russell (1925–1927).) It turns out that it is this scope distinction that is doing most of the work for Russell. And we can see this in his solutions to the three puzzles he sets forward to test theories of descriptions.

Example 20.1.1 (The Law of Excluded Middle) Here is Russell’s account:

By the law of excluded middle, either ‘A is B’ or ‘A is not B’ must be true. Hence either ‘the present King of France is bald’ or ‘the present King of France is not bald’ must be true. Yet if we enumerated the things that are bald, and then the things that are not bald, we should not find the present King of France in either list. Hegelians, who love a synthesis, will probably conclude that he wears a wig. (Russell, 1905, p. 48)

Russell’s solution makes explicit appeal to the scope distinction. Using our predicate abstraction notation rather than his notational devices, (20.1) is symbolized as

$$\langle \lambda x.B(x) \rangle (\iota x.K(x)), \quad (20.3)$$

where $\iota x.K(x)$ is understood informally to mean “the thing that is K ”. The sentence

$$\text{The present King of France is not bald,} \quad (20.4)$$

can be symbolized in two distinct ways:

$$\langle \lambda x. \neg B(x) \rangle (\iota x. K(x)) \quad (20.5)$$

$$\neg \langle \lambda x. B(x) \rangle (\iota x. K(x)) \quad (20.6)$$

(20.3) says of $\iota x. K(x)$ that he is bald; (20.5) says of $\iota x. K(x)$ that he is non-bald; (20.6) says it is not the case that $\iota x. K(x)$ is bald, or more briefly, that $\iota x. K(x)$ is not bald. (20.3) and (20.5) are both false because the term is non-designating; but (20.6) is true. Russell preserves the Law of Excluded Middle by distinguishing (20.5) from (20.6). (20.5) is not the contradictory of (20.3), but (20.6) is; and (20.3) and (20.6) in fact differ in truth value.

Example 20.1.2 (The Paradox of NonBeing) Russell's version is a bit baroque:

Consider the proposition 'A differs from B'. If this is true, there is a difference between A and B, which fact may be expressed in the form 'the difference between A and B subsists'. But if it is false that A differs from B, then there is no difference between A and B, which fact may be expressed in the form 'the difference between A and B does not subsist'. But how can a non-entity be the subject of a proposition? 'I think, therefore I am' is no more evident than 'I am the subject of a proposition, therefore I am', provided 'I am' is taken to assert subsistence or being, not existence. Hence, it would appear, it must always be self-contradictory to deny the being of anything; but we have seen, in connexion with Meinong, that to admit being also sometimes leads to contradictions. Thus if A and B do not differ, to suppose either that there is, or that there is not, such an object as 'the difference between A and B' seems equally impossible. (Russell, 1905, p. 48)

Russell's is the classic Deflationist answer to the Paradox. But his solution is misleading, for he assumes in his discussion that " x exists" is not a predicate. On Russell's view,

$$\text{The present King of France exists} \quad (20.7)$$

is a conjunction of the two claims:

At least one thing presently kings France, and

At most one thing presently kings France.

To deny (20.7) is to deny the conjunction, i.e., to deny that anything uniquely kings France at present:

$$\neg(\exists x)(K(x) \wedge (\forall y)(K(y) \supset x = y)). \quad (20.8)$$

Contrast (20.7) with (20.1). In the latter case we have three conjuncts; in the former case we have only two. There are two ways of negating (20.1): on the *narrow scope* reading of the definite description, one negates the conjunction of the three clauses; on the *wide scope* reading, one negates only the last of the three clauses. Since, in

the case of (20.7), there are only two clauses, the negation of (20.7) engenders no scope ambiguity.

There is no third conjunct in the expansion of (20.7) because Russell simply assumes “*x* exists” is not a predicate. This, as we have argued, is prejudicial. If we had a predicate “*x* exists,” then (20.7) would be understood to be expanded to three conjuncts

At least one thing presently kings France, and

At most one thing presently kings France, and

Whatever presently kings France exists.

And, in this case, there would be two ways of denying (20.7): either deny the conjunction of all three conjuncts, or simply deny the last conjunct. If we simply denied the last, we should be denying *of* the present King of France *that* he exists, and it is precisely this reading Russell sought to rule out.

Example 20.1.3 (The Puzzle About Identity) Russell, once more:

If *a* is identical with *b*, whatever is true of the one is true of the other, and either may be substituted for the other in any proposition without altering the truth or falsehood of that proposition. Now George IV wished to know whether Scott was the author of *Waverley*; and in fact Scott *was* the author of *Waverley*. Hence we may substitute *Scott* for *the author of ‘Waverley’*, and thereby prove that George IV wished to know whether Scott was Scott. Yet an interest in the law of identity can hardly be attributed to the first gentleman of Europe. (Russell, 1905, pp. 47–8)

Despite the truth of the identity,

Scott was the author of *Waverley*, (20.9)

substitution of one expression for the other in

George IV wished to know whether Scott was the author of *Waverley*, (20.10)

turns the true sentence into a false one:

George IV wished to know whether Scott was Scott. (20.11)

But, substituting coextensional terms in non-truth-functional contexts is not truth preserving. With a small scope reading, we cannot substitute for the description and preserve truth value: (20.11) does not follow from (20.9) and (20.10).

This last example shows very clearly that Russell had recognized as early as Russell (1905) that descriptions fail to act as proper names not just when they fail to designate, but when they occur inside non-truth-functional contexts (like *A believes that*). Indeed, it is just this character of Russell’s treatment of descriptions that

enabled Arthur Smullyan to seize upon the scope distinction as a structural way of marking the *de re/de dicto* distinction.¹

20.2 Definite Description Syntax

Formally there are two syntactic routes that could be followed with definite descriptions. On the one hand we could simply keep our official language to be what it has been in this book since Chap. 14, and consider definite descriptions to be abbreviations of more complex formulas. This would be similar to the way we might have taken negation and disjunction as primitive, considering formulas involving conjunction as abbreviations in the familiar way: $X \wedge Y$ abbreviates $\neg(\neg X \vee \neg Y)$, and so on. If we took this route, then a formula with a definite description in it should have it translated away before we begin trying to find a formal proof, or before considering semantic behavior.

There is a more direct route: rather than taking definite descriptions to be abbreviations that are translated away, one could think of them as primitive, and enlarge the language definition to incorporate them. This is the route we take, but either works quite well. Nonetheless, our choice is more complex than it appears at first glance. Definite descriptions contain formulas. But definite descriptions are treated like terms, so they can occur in formulas. Consequently it is no longer possible to define definite descriptions first and formulas afterward, nor is it possible to do it the other way around. The two must be defined together. The following replaces and broadens an earlier definition by adding a definite description case.

Definition 20.2.1 (First-Order Modal Formulas)

6. Definition 14.3.1, and ultimately Definition 8.1.2, is replaced with the following.

- a. A constant symbol c is an *intensional term*. It has no free variable occurrences.

¹ Smullyan (1948) calls our attention to the fact that theorem 14.18 of Whitehead and Russell (1925–1927),

$$E! \lambda x. \Phi x \supset [(\forall x) \Psi x \supset \Psi \lambda x. \Phi x],$$

which says that if the Φ exists, then we can substitute “the Φ ” for a variable, is further constrained by theorem 14.3,

$$\begin{aligned} ((\forall p)(\forall q)(p \equiv q \supset f(p) \equiv f(q)) \wedge E! \lambda x. \phi x) \supset \\ f\{[\lambda x. \phi x]. \xi \lambda x. \phi x\} \equiv [\lambda x. \phi x]. f\{\xi \lambda x. \phi x\} \end{aligned}$$

which says that scope does not matter if the Φ exists *and* f is truth-functional. Quine was apparently unaware of this constraint on substitution, for as late as (Quine, 1961, p. 154), he accuses Smullyan of “altering” Russell’s theory with this additional constraint, a charge excised in later editions.

- b. If f is an n -ary function symbol and $x_1, \dots, x_n$ are variables, $f(x_1, \dots, x_n)$ is an *intensional term*. Its free variable occurrences are the occurrences of $x_1, \dots, x_n$.
- c. If Ψ is a formula and y is a variable, $\iota y.\Psi$ is an intensional term, called a *definite description*. Its free variable occurrences are those of Ψ except for occurrences of y .
- d. For a formula Φ and a variable x , $\langle \lambda x.\Phi \rangle$ is a *predicate abstract*. The free variable occurrences of $\langle \lambda x.\Phi \rangle$ are those of Φ except for occurrences of x .
- e. If $\langle \lambda x.\Phi \rangle$ is a predicate abstract and t is an intensional term, then $\langle \lambda x.\Phi \rangle(t)$ is a formula. The free variable occurrences of this formula are the free variable occurrence of the predicate abstract $\langle \lambda x.\Phi \rangle$, together with the free variable occurrences of t .

As was said earlier, the notation $\iota y.\Psi$ is read “the y such that Ψ ,” and is called a *definite description*. If the definite description has no free variables then $\iota y.\Phi$ is used syntactically like an intensional constant. If it has free variables it is used syntactically like a (partial) intensional function. Either way, if $\iota y.\Psi$ is a definite description and $\langle \lambda x.\Phi \rangle$ is a predicate abstract, then $\langle \lambda x.\Phi \rangle(\iota y.\Psi)$ is a formula. This standard notation for definite descriptions, involving an upside down Greek iota, was introduced in Russell (1905) and Whitehead and Russell (1925–1927), though see Rodriguez-Consuegra (2000) for more detailed information.

Example 20.2.2 Here are two connected examples.

1. If $\mathcal{K}(y)$ is a full description of what it means for y to be the King of France, then $\iota y.\mathcal{K}(y)$ is read as “the y such that y is the King of France,” or more briefly just “the King of France.” If $\mathcal{B}(x)$ expresses that x is bald then $\langle \lambda x.\mathcal{B}(x) \rangle(\iota y.\mathcal{K}(y))$ expresses the proposition that the King of France is bald.
2. Continuing, suppose $\mathcal{M}(z, w)$ expresses the two place relation that z is the mother of w . Then $\langle \lambda w.\mathcal{M}(z, w) \rangle$ is the property of having z as mother, so $\langle \lambda w.\mathcal{M}(z, w) \rangle(\iota y.\mathcal{K}(y))$ is true just if z is the mother of the King of France. Then $\iota z.\langle \lambda w.\mathcal{M}(z, w) \rangle(\iota y.\mathcal{K}(y))$ is the mother of the King of France.

Exercises

Exercise 20.2.1 Show, using Definition 20.2.1, that the following is a term with no free variable occurrences (P is a one-place relation symbol):

$$\iota x.\langle \lambda y.x = y \rangle(\iota z.P(z)).$$

This term can be read: “the x that is equal to the z such that P .”

Exercise 20.2.2 Continue from part 2 of Example 20.2.2 and produce something expressing that the mother of the King of France is bald.

20.3 Semantics for Definite Descriptions

Assuming definite descriptions are primitive and not things to be translated away, how should they behave semantically? The choice of possibilist or actualist quantification is not critical—either will serve for the time being. The central issue is: what do we do when there is nothing for a definite description to describe?

One approach that is attractive at first glance is to say, “The King of France is bald” is neither true nor false since it refers to something that does not exist, namely the King of France. This leads to a partial, or three-valued logic, which is something we avoid here—see Blamey (1986) for a presentation of such an approach. It is not so simple to say that we will allow some formulas to lack a truth value; the question is “which ones?” and this is not always obvious. For instance, consider “Either the King of France is bald, or it is not the case that the King of France is bald.” We might want to take this as lacking a truth value because it refers to something that does not exist, or we might want to take this as true, since it has the form $\Phi \vee \neg\Phi$. Even more strongly, what about, “The King of France does not exist;” if it is not to be taken as true, how would we tell a Frenchman he is kingless? Fortunately it is not necessary to move to a partial logic in order to deal with the problem satisfactorily.

We can, in fact, avoid the whole issue of there being no King of France. Unlike Frege and Russell, we have all along allowed terms to designate nonexistent objects, and we have provided evaluation rules to assign truth values to sentences containing such terms. And, starting with Chap. 14, in particular Sect. 14.5, we have provided evaluation rules to assign truth values to sentences containing terms that lack designation altogether. We now simply extend these rules to definite descriptions. Here are the details.

In a modal model, an interpretation function is used to specify values for relation symbols, non-rigid constant symbols, and non-rigid function symbols. These values are arbitrarily specified, and constitute part of the definition of the model. From here on we will also make use of interpretation functions to handle definite descriptions, but with the important difference that their values are not arbitrary specifications, but are calculated. These values arise from the model itself, and are not things we put in directly. The following essentially extends Definition 14.6.1.

Definition 20.3.1 (Interpretations Extended to Definite Descriptions) Let $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ be a model (constant or varying domain). We extend the definition of the interpretation function $\mathcal{I}$ to definite descriptions as follows. Let $\iota x.\Phi$ be a definite description. We say $\mathcal{I}(\iota x.\Phi, \Gamma)$ is defined in $\mathcal{M}$ under valuation v if there is exactly one x -variant w of v such that $\mathcal{M}, \Gamma \Vdash_w \Phi$; and if this is the case then $\iota x.\Phi$ *designates* at Γ under v , and specifically, $\mathcal{I}(\iota x.\Phi, \Gamma) = w(x)$.

The idea behind this should be obvious. A definite description $\iota x.\Phi$ formally designates at a world in a model if exactly one value of x makes Φ true, and if this happens, that value is what it designates. It should be noted that in the Definition above, the valuation v and an x -variant w of v may assign any values in the domain of the model to x . It is not required that they be members of $\mathcal{D}(\Gamma)$. Also if Φ

contains no free variables other than x , whether $\iota x.\Phi$ designates, and what, are independent of a choice of valuation. In such a case we will simply say that $\iota x.\Phi$ designates and omit any mention of a valuation, and similarly for what it designates.

The kind of difficulties that arose when we discussed syntax for definite descriptions arise here too. In order to determine whether $\iota x.\Phi$ designates at a possible world Γ , and to determine its denotation if it does, we need to check whether Φ is true at Γ under various valuations. But to check the truth of a formula at a possible world we need to know about the denotations of terms that appear in the formula, and Φ could contain additional definite descriptions which must be evaluated. The situation is circular in the sense that we can no longer define designations for terms first, and then truth at worlds afterwards. Both must be defined together. Of course, what makes all this work is that each step of term evaluation or of truth-value determination involves a reduction in syntactic complexity, and so the circular references “bottom out” eventually. We skip the formal details and rely on your intuition that this is so.

We have sidestepped entirely the existence issue, which historically has been pretty much at the forefront of the discussion of definite descriptions. In doing so, we have adopted Frege’s semantic idea that definite descriptions are singular terms, and combined it with Russell’s scope distinction which enables us to capture the variability of designation—and even the effect of their nondesignation. Frege’s intuitions are central in our model theoretic account above of the behavior of descriptions, but Russell’s ideas will resurface. In Sect. 20.5, we will show how Russell’s account can be embedded in a Fregean account.

Now that we have seen how to treat definite descriptions as non-rigid designators, extending the interpretation function to them, we can simply apply earlier machinery to handle their semantic behavior as parts of formulas. Definition 14.6.2 has one more part added to it, as follows.

Definition 20.3.2 Continuing Definition 14.6.2:

12. For a definite description $\iota x.\Phi$:

- a. If $\iota x.\Phi$ designates at Γ in $\mathcal{G}$ under valuation v (Definition 20.3.1) then $\mathcal{M}, \Gamma \Vdash_v \langle \lambda y.\Psi \rangle (\iota x.\Phi) \iff \mathcal{M}, \Gamma \Vdash_w \Psi$, where w is the y -variant of v such that $w(y) = \mathcal{I}(\iota x.\Phi, \Gamma)$.
- b. If $\iota x.\Phi$ fails to designate at Γ under v then $\mathcal{M}, \Gamma \nVdash_v \langle \lambda y.\Phi \rangle (\iota x.\Phi)$.

20.4 Some Examples

Is the present King of France King of France? Is the flying horse of Greek mythology a horse? And is the round square a square? We find ourselves in agreement with Whitehead and Russell (1925–1927), who observed:

... such a proposition as “the man who wrote *Waverley* wrote *Waverley*” does not embody a logically necessary truth, since it would be false if *Waverley* had not been written, or had

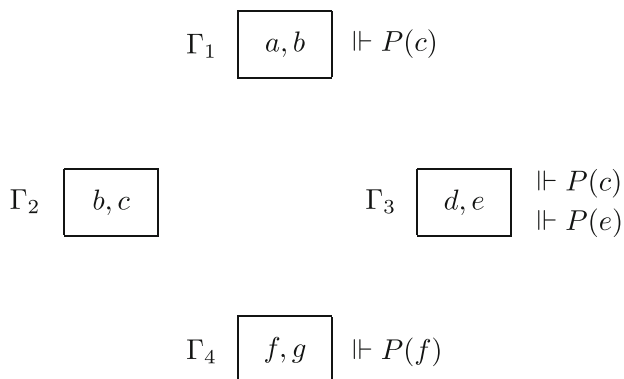
been written by two men in collaboration. For example, “the man who squared the circle squared the circle” is a false proposition.

The sentence “The present King of France is king of France” is not true in this world, because there is no French king now. The round square case is different. The defining property of round squares is contradictory, and so round squares can exist in no possible world of any model. Consequently it is false at every world that the round square is square, since the definite description can never designate.

For this section, assume semantics is varying domain, because we wish to consider the roles of existence and nonexistence. Our distinction between designation and existence is of great importance in understanding the semantics of definite descriptions. The description “the first President of the United States” designates George Washington. The definite description designates at the present time, but it does not designate something that exists at the present time. By contrast, the definite description, “the present King of France,” does not even designate now. It is easy to overlook this distinction because existence assumptions, as opposed to designation assumptions, are frequently tacitly built into a definite description. For example, “the tallest person in the world,” takes for granted that we mean to designate an existent person. By way of contrast, “the tallest person in the world in 1995” presumably also designates, but it is contingent whether the person designated currently exists or not—it is not an assumed condition.

In the present section we will explain how definite descriptions operate by looking at examples, beginning more formally, and ending up, hopefully, in accordance with colloquial usage.

Example 20.4.1 Suppose we have a varying domain model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ with $\mathcal{G}$ containing exactly four possible worlds, with the worlds accessible from each other, including from themselves. With this understood, we don’t bother to represent the accessibility relation explicitly. It is an **S5** model, though this plays no special role. Suppose further that the domains associated with the four possible worlds, and the truth-value at worlds for atomic formulas involving the one-place relation symbol P are given according to the following schematic diagram.



In this model, $\iota x.P(x)$ designates at Γ_4 , and in fact it designates f there. We check this claim in detail, and skip the formalities for later such assertions.

Let v be any valuation, and consider $\mathcal{J}(\iota x.P(x), \Gamma_4)$ under v . Obviously if v' is the x -variant of v such that $v'(x) = f$, we have that $\mathcal{M}, \Gamma_4 \Vdash_{v'} P(x)$. But also, if v'' is any x -variant of v besides v' , so that $v''(x) \neq f$, then $\mathcal{M}, \Gamma_4 \nVdash_{v''} P(x)$. Then according to Definition 16.1.3, $\iota x.P(x)$ designates at Γ_4 , and $\mathcal{J}(\iota x.P(x), \Gamma_4) = v'(x) = f$.

Since $\mathcal{J}(\iota x.P(x), \Gamma_4)$ is defined, we have that $\mathbf{D}(\iota x.P(x))$ is true at Γ_4 , where we have used the designation predicate abstract of Definition 18.1.1. And further, since f is in the domain of Γ_4 we also have that $\mathbf{E}(\iota x.P(x))$ is true at Γ_4 .

At Γ_1 , $\iota x.P(x)$ also designates—here it designates c . Thus $\mathbf{D}(\iota x.P(x))$ is true at Γ_1 . But since c is not in the domain of Γ_1 we have $\bar{\mathbf{E}}(\iota x.P(x))$ at Γ_1 .

At Γ_2 , $\iota x.P(x)$ does not designate, because P is true of nothing at Γ_2 . Likewise at Γ_3 , $\iota x.P(x)$ does not designate, but now because there is more than one thing that P is true of at Γ_3 .

Example 20.4.2 Formulas of the form $\langle \lambda x.\Phi \rangle(\iota x.\Phi)$ are not always valid, because they are false if $\iota x.\Phi$ does not designate, though things change if designation is ensured. The following *is* valid.

$$\mathbf{D}(\iota x.\Phi) \supset \langle \lambda x.\Phi \rangle(\iota x.\Phi)$$

It is not difficult to check the validity of this. Suppose we have a model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{J} \rangle$ and, for $\Gamma \in \mathcal{G}$ and valuation v , we have

$$\mathcal{M}, \Gamma \Vdash_v \mathbf{D}(\iota x.\Phi)$$

or equivalently,

$$\mathcal{M}, \Gamma \Vdash_v \langle \lambda x.x = x \rangle(\iota x.\Phi).$$

Then $\iota x.\Phi$ must designate at Γ with respect to v , so according to Definition 20.3.1 there is exactly one x -variant, v' of v such that $\mathcal{M}, \Gamma \Vdash_{v'} \Phi$, and $\mathcal{J}(\iota x.\Phi) = v'(x)$. But then by definition of satisfiability for predicate abstracts,

$$\mathcal{M}, \Gamma \Vdash_v \langle \lambda x.\Phi \rangle(\iota x.\Phi).$$

As a matter of fact, the converse implication is also valid. We thus have a central fact—the following is valid:

$$\mathbf{D}(\iota x.\Phi(x)) \equiv \langle \lambda x.\Phi(x) \rangle(\iota x.\Phi(x)) \quad (20.12)$$

Example 20.4.3 Think of the real world and its conceivable alternatives, including the world of Greek mythology, as constituting a modal model. Take the model as varying domain—quantifiers are actualist. By the equivalence (20.12) above, the

flying horse of Greek mythology is a horse provided the term designates. In fact, Pegasus does not exist, but in the world of Greek mythology it does. Thus, if $P(x)$ is a one-place relation symbol intended to characterize the flying horse the Greeks spoke of, the following is true at the real world of our informal model:

$$\bar{\mathbf{E}}(\iota x.P(x)) \wedge \Diamond \mathbf{E}(\iota x.P(x)) \wedge \mathbf{D}(\iota x.P(x))$$

One might be uneasy with this modeling of our intuition. For if Pegasus is a horse one might feel obligated to include treatment of Pegasus in the study of biology. An alternative modeling takes the description to be:

the unique x such that
it is said that (x flies and x is a horse)

The description designates, for the x such that it is said to fly and be a horse *is* said to fly and be a horse. But we don't have to worry about the biology of such a creature.

Example 20.4.4 Finally we have an example that really involves modal notions. Consider the sentence, “Someday, somebody could be taller than the tallest person in the world.” Suppose we formulate this in a modal logic in which possible worlds are intended to be temporal states of the world, reading $\Box\Phi$ as asserting that Φ is and always will be the case. Take the model to be varying domain. Let $T(x)$ be a one-place relation symbol that characterizes the tallest person in the world—the intention is that $T(x)$ should be true of an object precisely when that object is a person, that person currently exists, and all other currently existing people are shorter. Now, here is a formalization of “Someday, somebody could be taller than the tallest person in the world.”

$$\langle \lambda x. \Diamond \neg \langle \lambda y. x = y \rangle (\iota z. T(z)) \rangle (\iota z. T(z)) \quad (20.13)$$

In order to read this properly, first note that $\langle \lambda y. x = y \rangle$ is the “same as x ” predicate, so the subformula $\neg \langle \lambda y. x = y \rangle (\iota z. T(z))$ says the tallest person in the world is not the same as x . Combining this with the rest of the formula, then, we have: “It is true of the tallest person in the world (now) that, someday, the tallest person in the world will not be that person.”

Sentence (20.13) is satisfiable, but we can do better than that. Let us make some reasonable assumptions about our notions. First, existence is an inherent assumption about “the tallest person in the world.” Let us make this explicit with $\mathbf{E}(\iota x. T(x))$. Second, let us postulate that any given person might not be the tallest in the world someday, whether or not they are now: $(\forall x) \Diamond \neg T(x)$. From these formula (20.13) follows. That is, the following is valid in the simplest modal logic we have considered, varying domain **K**:

$$[\mathbf{E}(\iota x. T(x)) \wedge (\forall x) \Diamond \neg T(x)] \supset \langle \lambda x. \Diamond \neg \langle \lambda y. x = y \rangle (\iota z. T(z)) \rangle (\iota z. T(z)) \quad (20.14)$$

This is actually not hard to see informally, but we give a formal verification so you can see how the machinery works. We show that whenever the antecedent holds, so does the consequent.

Assume $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ is a varying domain **K** model, $\Gamma \in \mathcal{G}$, and the antecedent holds at Γ . That is, we have the following.

$$\mathcal{M}, \Gamma \Vdash_v \mathbf{E}(\iota x.T(x)) \wedge (\forall x)\Diamond \neg T(x) \quad (20.15)$$

Using Definitions 15.4.3 and 12.1.2, the first of the conjuncts gives us

$$\mathcal{M}, \Gamma \Vdash_v \langle \lambda x.(\exists y)(y = x) \rangle (\iota x.T(x)) \quad (20.16)$$

and since this predicate abstract is the case it follows that $\iota x.T(x)$ designates at Γ under v . Further, what it designates must be something making $(\exists y)(y = x)$ true at Γ . Since we are using varying domain semantics, this must be something that is in the domain of Γ . We conclude that $\iota x.T(x)$ designates at Γ something that is in the domain of Γ ; suppose we say this thing is d .

Since the second conjunct of (20.15) is true at Γ , and d is in the domain of Γ , we have

$$\mathcal{M}, \Gamma \Vdash_{v'} \Diamond \neg T(x) \quad (20.17)$$

where v' is the x variant of v such that $v'(x) = d$. Then there must be some $\Delta \in \mathcal{G}$ with $\Gamma \mathcal{R} \Delta$ so that we have the following.

$$\mathcal{M}, \Delta \Vdash_{v'} \neg T(x) \quad (20.18)$$

Now we show that at Γ the consequent of (20.14) must hold, that is, we show we have the following.

$$\mathcal{M}, \Gamma \Vdash_v \langle \lambda x.\Diamond \neg \langle \lambda y.x = y \rangle (\iota z.T(z)) \rangle (\iota z.T(z)) \quad (20.19)$$

Since we know that $\iota z.T(z)$ designates d at Γ , this is equivalent to the following.

$$\mathcal{M}, \Gamma \Vdash_v \Diamond \neg \langle \lambda y.x = y \rangle (\iota z.T(z)) \quad (20.20)$$

To show that we have this, it would be enough to show we have

$$\mathcal{M}, \Delta \Vdash_{v'} \neg \langle \lambda y.x = y \rangle (\iota z.T(z)) \quad (20.21)$$

and we do have this for the following reasons. It may be that $\iota z.T(z)$ does not designate at Δ , in which case no predicate abstract applies to it, and in particular $\langle \lambda y.x = y \rangle (\iota z.T(z))$ is false at Δ . Or it may be that $\iota z.T(z)$ does designate at Δ , but what it designates cannot be d because T does not apply to d at Δ by (20.18) but $v'(y)$ is d , so again $\langle \lambda y.x = y \rangle (\iota z.T(z))$ is false at Δ .

There is a little more of interest that can be extracted from this example. To begin, sentence (20.13) is trivially equivalent to

$$\langle \lambda x. \neg \Box \langle \lambda y. x = y \rangle (\iota z. T(z)) \rangle (\iota z. T(z)).$$

We leave it to you to check that, even for terms t that can fail to designate, $\langle \lambda x. \neg \Phi \rangle (t) \supset \neg \langle \lambda x. \Phi \rangle (t)$ is valid. So a consequence of (20.13) is

$$\neg \langle \lambda x. \Box \langle \lambda y. x = y \rangle (\iota z. T(z)) \rangle (\iota z. T(z)).$$

Now, Proposition 19.2.7 (extended from constant symbols to definite descriptions) says this is the negation of the assertion that $\iota z. T(z)$ is rigid. (Actually, Proposition 19.2.7 was proved under the assumption that terms always designated, but designation is a consequence of the existence assumption we made above about $\iota z. T(z)$.) Non-rigidity should be an “obvious” consequence, since if $\iota z. T(z)$ could not vary its designation, the tallest person in the world could never change.

Finally we have been using varying domain models, but if we had used a constant domain semantics things would have been simpler. In a constant domain setting the condition $E(\iota x. T(x))$ in (20.14) would not have been needed.

Example 20.4.5 Consider the description “the present King of France.” There is no present King of France (in this world) and so if the term designated, it would not designate an existing object. But it does not even designate, because the predicate “ x presently kings France” is true uniquely for no value of the variable. The present King of France fails to exist; and the term “the present King of France” fails to designate. Of course, there are other worlds in which it does designate, and indeed, is an object existing in that world. We might put the point this way:

$$\langle \lambda x. \Diamond E x \rangle (\iota x. K(x))$$

is false; but

$$\Diamond \langle \lambda x. E x \rangle (\iota x. K(x))$$

is true.

Example 20.4.6 Here are a few more Russell-like examples, in addition to “George IV didn’t know that Scott was the author of Waverley.”

Let $W(z)$ be a formula asserting that z is the author of Waverley. Consider the sentence: “George IV didn’t know the author of Waverley was the author of Waverley.” More precisely worded, it is, “George IV didn’t know, of the author of Waverley, that he was the author of Waverley.” Suppose we read $\Box$ epistemically as “George IV knows that. . .” Then this formalizes directly as the following.

$$\langle \lambda x. \neg \Box \langle \lambda y. x = y \rangle (\iota z. W(z)) \rangle (\iota z. W(z))$$

In addition, we could also represent it as the following less complex formula.

$$\langle \lambda x. \neg \Box W(x) \rangle (\iota z. W(z))$$

We give as exercises below to show that these are satisfiable.

Next sentence: “George IV knew there was an author of Waverley, but he didn’t know who it was.” This formalizes as follows.

$$\Box E(\iota z. W(z)) \wedge (\forall w) \neg \Box \langle \lambda y. w = y \rangle (\iota z. W(z))$$

We also give the satisfiability of this as an exercise.

Finally, to continue with the previous sentence, the following is not an adequate formalization of “George IV knew there was an author of Waverley, but he didn’t know who it was.”

$$\Box E(\iota z. W(z)) \wedge (\forall w) \langle \lambda y. \neg \Box w = y \rangle (\iota z. W(z))$$

This is *not* satisfiable—in fact, its negation is valid. We give this as an exercise once tableau rules have been introduced.

Example 20.4.7 Our final example involves the converse of the formula in Exercise 20.4.4. These formulas will be revisited again once we have proof methods available for definite descriptions. For a one-place predicate symbol P , the formula $\mathbf{D}(\iota x. \Diamond P(x)) \supset \Diamond \mathbf{D}(\iota x. P(x))$ is valid (in varying domain $\mathbf{K}$). To verify this, assume we have a model $\mathcal{M} = \langle \mathcal{D}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ and a possible world Γ in it such that $\mathcal{M}, \Gamma \Vdash_v \mathbf{D}(\iota x. \Diamond P(x))$; we show it follows that $\mathcal{M}, \Gamma \Vdash_v \Diamond \mathbf{D}(\iota x. P(x))$.

By our assumption, $\mathcal{M}, \Gamma \Vdash_v \langle \lambda y. (y = \iota x. \Diamond P(x)) \rangle (\iota x. \Diamond P(x))$. For this to happen, $\iota x. \Diamond P(x)$ must designate at Γ (under valuation v), so for exactly one x -variant w of v we have $\mathcal{M}, \Gamma \Vdash_w \Diamond P(x)$.

Then there must be some $\Delta \in \mathcal{G}$ with $\Gamma \mathcal{R} \Delta$ and $\mathcal{M}, \Delta \Vdash_w P(x)$. We claim that w is the *only* x variant of v such that $\mathcal{M}, \Delta \Vdash_w P(x)$. For suppose u is an x variant of v (and hence also of w) such that $u(x) \neq w(x)$, and $\mathcal{M}, \Delta \Vdash_u P(x)$. Since $\Gamma \mathcal{R} \Delta$, $\mathcal{M}, \Gamma \Vdash_u \Diamond P(x)$. But w is the *only* x -variant of v such that $\mathcal{M}, \Gamma \Vdash_w \Diamond P(x)$, while u and w are different x -variants of v , and this is impossible.

Since w is the unique x -variant of v such that $\mathcal{M}, \Delta \Vdash_w P(x)$, then $\iota x. P(x)$ designates at Δ under v , and so $\mathcal{M}, \Delta \Vdash_v \mathbf{D}(\iota x. P(x))$. But then $\mathcal{M}, \Gamma \Vdash_v \Diamond \mathbf{D}(\iota x. P(x))$, as desired.

Exercises

Exercise 20.4.1 Let P be a one-place relation symbol. Give an explicit model showing that $\langle \lambda x. P(x) \rangle (\iota x. P(x))$ is not valid.

Exercise 20.4.2 Show the validity of

$$\langle \lambda x. \Psi(x) \rangle (\iota x. \Phi(x)) \supset \mathbf{D}(\iota x. \Phi(x)).$$

Then give a varying domain model showing the non-validity of

$$\langle \lambda x. \Psi(x) \rangle (\iota x. \Phi(x)) \supset \mathbf{E}(\iota x. \Phi(x)).$$

Exercise 20.4.3 Show the satisfiability of the following sentences, all taken from Example 20.4.6.

1. $\langle \lambda x. \neg \Box \langle \lambda y. x = y \rangle (\iota z. W(z)) \rangle (\iota z. W(z))$
2. $\langle \lambda x. \neg \Box W(x) \rangle (\iota z. W(z))$
3. $\Box \mathbf{E}(\iota z. W(z)) \wedge (\forall z) \neg \Box \langle \lambda y. z = y \rangle (\iota z. W(z))$

Exercise 20.4.4 Is the formula $\Diamond \mathbf{D}(\iota x. P(x)) \supset \mathbf{D}(\iota x. \Diamond P(x))$ valid or invalid in varying domain **K**? (This is the converse of the formula from Example 20.4.7.) Semantically establish the correctness of your answer.

Exercise 20.4.5 Show the validity in varying domain **K** of the following, generalizing Exercise 20.4.4:

$$\langle \lambda x. \Box A(x) \rangle (\iota x. \Diamond P(x)) \supset \Diamond \langle \lambda x. A(x) \rangle (\iota x. P(x)).$$

Exercise 20.4.6 Just as we defined existence and nonexistence properties, we can define possible-existence and necessary-existence properties.

$$\begin{aligned} \mathbf{E}_{\Box}(t) &= \langle \lambda x. \Box \mathbf{E}(x) \rangle (t) = \langle \lambda x. \Box (\exists y)(y = x) \rangle (t) \\ \mathbf{E}_{\Diamond}(t) &= \langle \lambda x. \Diamond \mathbf{E}(x) \rangle (t) = \langle \lambda x. \Diamond (\exists y)(y = x) \rangle (t) \end{aligned}$$

Use the model of Example 20.4.1 and determine at which worlds of it the sentence $\mathbf{E}_{\Diamond}(\iota x. P(x))$ is true; and similarly for $\Diamond \mathbf{E}(\iota x. P(x))$ is true. Do the same with $\mathbf{E}_{\Box}(\iota x. P(x))$ and $\Box \mathbf{E}(\iota x. P(x))$.

Exercise 20.4.7 Using the notation of Exercise 20.4.6, determine the relationship between $\mathbf{E}_{\Diamond}(t)$ and $\Diamond \mathbf{E}(t)$, where t is a definite description. That is, does the first always imply the second, or not, and similarly the other way around. Also do the same for the pair $\mathbf{E}_{\Box}(t)$ and $\Box \mathbf{E}(t)$.

Exercise 20.4.8 One ontological argument for the existence of God hinges on making necessary existence part of the defining condition. Consider the necessarily existent being, $\iota x. \mathbf{E}_{\Box}(x)$. Discuss the relationships between assertions that this term denotes, that the being denoted has existence, and that the being has necessary existence.

Exercise 20.4.9 Consider the description “the possible fat man in the doorway.” Construct three distinct formal representations of this description and discuss how they differ.

20.5 Russell's Approach

In Sect. 20.1 we discussed two quite different ways of dealing with definite descriptions. Following ideas stemming from Frege and Carnap, they could be treated as formal singular terms; we turn to this idea beginning in Sect. 20.7. But here we adopt Russell's approach which is, roughly, to treat them as not really being there. They should be paraphrased away. Whatever one thinks of this as a philosophical idea, formally it works quite well, but with one catch which we will come to shortly.

Recall Russell held that though definite descriptions function syntactically like terms, semantically they are myths. He shifted the question from what "the present King of France" designates to how it behaves in context. According to his theory, to say "The present King of France is bald" is to say: "there is one and only one object that meets the conditions for being the King of France, and that object is also bald." Definite descriptions simply disappear. Russell's version, developed in the framework of classical logic, extends to modal logic quite well. We can think of it as a translation device, paraphrasing away all definite description occurrences, in context.

Definition 20.5.1 (Russell Paraphrase) What we call the *Russell paraphrase* of $\langle \lambda x. \Psi(x) \rangle (\iota y. \Phi(y))$ is the following formula. (It is assumed that z is a variable that does not occur in $\Phi(x)$, and x is substitutable for y in $\Phi(y)$. If this should not be the case, one can rename variables using new letters.)

$$(\exists x) \{ \Phi(x) \wedge \\ (\forall z) [\Phi(z) \supset (z = x)] \wedge \\ \Psi(x) \\ \}$$

In a Russell paraphrase each separate part plays a well-defined role: there is something such that Φ ; nothing else is such that Φ ; and that thing is also such that Ψ . It can sometimes be convenient to give this paraphrase in an equivalent but more compact form, which we may also call Russell's Paraphrase:

$$(\exists x) \{ (\forall z) [\Phi(z) \equiv (z = x)] \wedge \Psi(x) \}.$$

A treatment of definite descriptions based on Russell's ideas is simple. Before evaluating a formula at a world of a model, or attempting to prove it using tableaux, use Russell's paraphrase to remove all definite description occurrences, and evaluate the result instead.

For constant domains, the use of Russell's Paraphrase as just described works very well. It is not hard to see that in a constant domain setting, the Paraphrase for a definite description corresponds directly to the semantic conditions of Definitions 20.3.1 and 20.3.2. This is not the case with varying domain models. For instance, when regarding the semantic behavior of $\iota x. \Phi$ at possible world Γ in

Definition 20.3.1, there is no requirement that valuations assign to x something in the domain of Γ if we have a varying domain model. But then it would not be something quantifiers could quantify over and the Russell Paraphrase will not work as intended.

Example 20.5.2 In Example 20.4.7 we argued semantically for the validity of $\mathbf{D}(\lambda x.\Diamond P(x)) \supset \Diamond \mathbf{D}(\lambda x.P(x))$ in varying domain $\mathbf{K}$. We now use tableaux to establish provability via the Russell embedding. But in accordance with the observation above, we switch from varying to constant domain because of the problems with the quantifiers. The formula, unabbreviated, is $\langle \lambda y.y = y \rangle(\lambda x.\Diamond P(x)) \supset \Diamond \langle \lambda y.y = y \rangle(\lambda x.P(x))$ and its Russell paraphrase is the following.

$$\begin{aligned} &(\exists x)[\Diamond P(x) \wedge (\forall y)(\Diamond P(y) \supset y = x) \wedge (x = x)] \supset \\ &\Diamond(\exists x)[P(x) \wedge (\forall y)(P(y) \supset y = x) \wedge (x = x)] \end{aligned}$$

To keep the clutter down as far as possible, we omit the $(x = x)$ subformulas, which behave like conjuncts that are always true and so can be ignored. Thus the formula to be proved using tableaux in $CN \mathbf{K}$ is the following.

$$\begin{aligned} &(\exists x)[\Diamond P(x) \wedge (\forall y)(\Diamond P(y) \supset y = x)] \supset \\ &\Diamond(\exists x)[P(x) \wedge (\forall y)(P(y) \supset y = x)] \end{aligned}$$

The tableau proof is shown in Fig. 20.1, and the steps are justified as follows. Lines 2 and 3 are from 1 by an implication rule; 4 is from 2 by an existential rule; 5 and 6

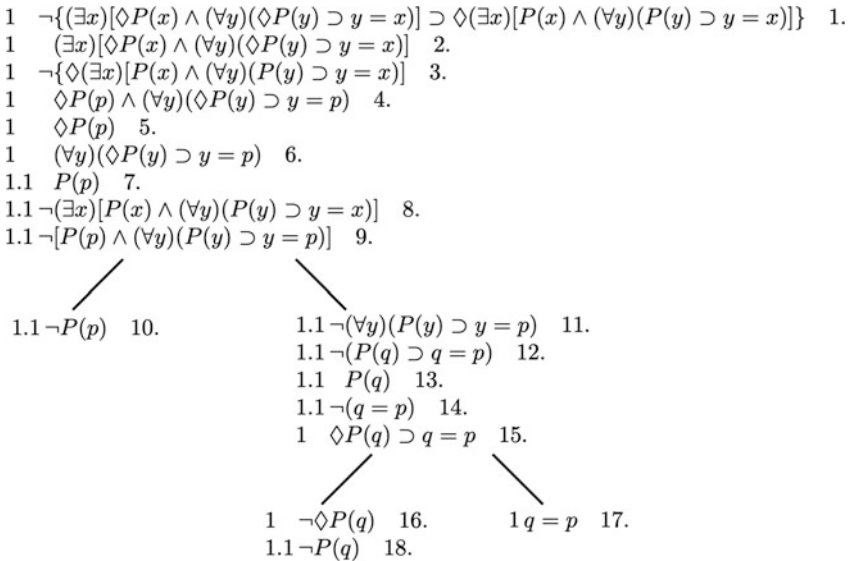


Fig. 20.1 Proof using the Russell embedding

are from 4 by conjunction; 7 is from 5 by a possibility rule; 8 is from 3 by necessity; 9 is from 8 by a universal rule; 10 and 11 are from 9 by disjunction; 12 is from 11 by an existential rule; 13 and 14 are from 12 by negated implication; 15 is from 6 by a universal rule; 16 and 17 are from 15 by an implication rule, and 18 is from 16 by a necessity rule.

The left branch is closed by 7 and 10. The middle branch is closed by 13 and 18. The right branch is closed by 14 and 17.

It should not be surprising that the Russell Paraphrase gives incompatible results when varying domain semantics is employed. Russell's explicit use of the existential quantifier imposes existence assumptions, as opposed to the designation assumptions that our varying domain semantic conditions are based on.

Definition 20.5.3 (Existence Supposition) We say a definite description *makes an existence supposition* if it is of the form $\iota x.(E(x) \wedge \Phi)$.

Proposition 20.5.4 *The following are equivalent in varying domain models:*

1. $\langle \lambda x.\Phi(x) \rangle(\iota y.E(y) \wedge \Psi(y))$.
2. *The Russell Paraphrase of $\langle \lambda x.\Phi(x) \rangle(\iota y.E(y) \wedge \Psi(y))$.*
3. *The Russell Paraphrase of $\langle \lambda x.\Phi(x) \rangle(\iota y.\Psi(y))$.*

We leave the proof of this as an exercise. Essentially, it says that the Russell Paraphrase and our semantic approach do coincide in varying domain models for those definite descriptions that involve an existence assumption. Still, an existence assumption is not something we can always count on.

We conclude this section with a curious observation due to Saul Kripke, Kripke (2011). Russell's device of treating a definite description as an abbreviation for a more complex construction is very natural, and very common in the literature, but things are not quite as straightforward as one might think.

The massive *Principia Mathematica* appeared in two editions, Whitehead and Russell (1910–1913) and Whitehead and Russell (1925–1927). One of the points on which the two editions differed was the choice of the primitive propositional logical connectives. The first edition used negation and disjunction, while the second used a single connective, the Sheffer stroke, “not-both” (see Exercise 2.7.1). Russell thought this an extremely important change. But he didn't realize it had unfortunate consequences where definite descriptions were concerned, because it introduces duplications that are easy to miss taking into account. For instance, using the Sheffer stroke $\neg A$ is really $A \uparrow A$. Then if A contains a definite description, it is really present at least twice in $\neg A$. Kripke showed that if the Sheffer stroke is the only propositional primitive, then there are definite descriptions for which some definite description elimination path, translating them away, does not terminate. Things can be tricky. We have chosen to take all the standard connectives as primitive and, fortunately, for this situation infinite reduction paths cannot occur.

Exercises

Exercise 20.5.1 Give a proof of Proposition 20.5.4.

Exercise 20.5.2 Show the equivalence of the two versions of Russell's paraphrase:

1. $(\exists x)\{(\forall y)[\Phi(y) \equiv (y = x)] \wedge \Psi(x)\};$
2. $(\exists x)\{\Phi(x) \wedge (\forall y)[\Phi(y) \supset (y = x)] \wedge \Psi(x)\}.$

Exercise 20.5.3 Example 20.4.2 has the following more specialized version. Suppose definite descriptions are treated using Russell's paraphrase (or equivalently, that they make an existential assumption). Under this assumption, show the validity of

$$\mathbf{E}(\lambda x.\Phi(x)) \equiv \langle \lambda x.\Phi(x) \rangle (\lambda x.\Phi(x)).$$

Exercise 20.5.4 Give a constant domain model in which the sentence $\Diamond \mathbf{D}(\lambda x.P(x)) \supset \mathbf{D}(\lambda x.\Diamond P(x))$ is not valid.

Exercise 20.5.5 Use constant domain tableaux to prove the relativization of

$$\mathbf{D}(\lambda x.\Diamond P(x)) \supset \Diamond \mathbf{D}(\lambda x.P(x))$$

to an existence predicate $\mathcal{E}$. This amounts to verifying that the formula holds in all varying domain models.

20.6 Our Strong Recommendations

Experience and a desire for simplicity lead us to give some important advice.

Strong Recommendation 20.6.1 *When definite descriptions are involved, choose constant domain tableau systems.*

Generally speaking, constant domains are the nicest both semantically and proof theoretically. They also tend to be the most natural in many settings. But what if, for some reason, we really want varying domains.

Strong Recommendation 20.6.2 *When desired, simulate varying domains by incorporating an explicit existence predicate, $\mathcal{E}$, into the language, as we did semantically in Sect. 8.9, and with tableaux in Sect. 12.10.*

The use of an existence predicate with constant domains allows us to quantify over an entire model domain, while retaining the ability to express the kinds of distinctions that actualist quantifiers are meant to capture, via relativized quantifiers. You were asked to do an exercise along these lines, Exercise 20.5.5. The Recom-

recommendations above apply to tableaux for definite descriptions whether the Russell Paraphrase is used, or the direct tableau rules for definite descriptions given below in Sect. 20.7.

20.7 Tableaux for Definite Descriptions

Instead of using the Russell paraphrase, one can apply tableau rules designed specifically for definite descriptions. We present such rules in this section. Following our Strong Recommendation 20.6.1, we give the rules for *constant domain* modal logics only. As we noted in Strong Recommendation 20.6.2, if varying domains should be wanted an existence predicate $\mathcal{E}$ can be used, so nothing is lost. Our rules, then, are extensions in the *CN* family: domains are constant but function and constant symbols, may not always designate. The underlying modal logic can be anything from the restricted modal cube—we will only discuss the details for **K**, leaving the rest to you.

To begin, we note that a definite description is a particular kind of non-rigid designator and it can fail to designate under some circumstances. What distinguishes a definite description from a term involving constant and function symbols is that designation behavior is not something that is directly specified semantically, as part of the definition of an interpretation, but rather it is calculated from the formula that appears in the definite description. But still, since a definite description is a non-rigid designator, we can extend to it the superscripting convention described at the beginning of Sect. 15.2.

Definition 20.7.1 (Object Term, Extended) This extends Definition 15.2.1. If $\lambda x.\Phi$ is a definite description and σ is a prefix, then $[\lambda x.\Phi]^\sigma$ is an object term. As with other object terms, it is *pseudo-closed* if it contains no free variables other than parameters.

Informally $[\lambda x.\Phi]^\sigma$ is intended to be the object that $\lambda x.\Phi$ denotes at possible world σ . With the definition of object term now extended, the machinery of earlier tableaux is similarly extended. In particular, Substitutivity of Equality applies to object terms involving definite descriptions, Definition 15.4.1. The notion of positive or negative generation now has an added clause to cover definite descriptions.

Definition 20.7.2 (Positively and Negatively Generated, Extended) This extends Definition 15.3.1.

3. For a definite description $\lambda x.\Psi$ we say:

- a. $[\lambda x.\Psi]^\sigma$ is *positively generated* on a tableau branch if the branch contains $\sigma \langle \lambda x.\Phi(x) \rangle (\lambda x.\Psi)$ for some $\Phi(x)$;
- b. $[\lambda x.\Psi]^\sigma$ is *negatively generated* on a tableau branch if the branch contains $\sigma \neg \langle \lambda x.\Phi(x) \rangle (\lambda x.\Psi)$ for some $\Phi(x)$.

Next, the *CN* rules given in Definitions 15.4.12 and 15.4.13 are extended to definite descriptions. We state the extensions explicitly for convenience.

Definition 20.7.3 (Predicate Abstraction Rules, *CN*) These rules extend Definition 15.4.12. For definite description $\iota y.\Psi(y)$:

$$\frac{\sigma \langle \lambda x.\Phi(x) \rangle (\iota y.\Psi(y))}{\sigma \Phi([\iota y.\Psi(y)]^\sigma)} \qquad \frac{\sigma \neg \langle \lambda x.\Phi(x) \rangle (\iota y.\Psi(y))}{\sigma \neg \Phi([\iota y.\Psi(y)]^\sigma)} \text{ provided } [\iota y.\Psi(y)]^\sigma \text{ is positively generated on the tableau branch}$$

Definition 20.7.4 (Existence Rule, *CN*) This extends Definition 15.4.13. For definite description $\iota y.\Psi(y)$:

$$\frac{\sigma \langle \lambda x.\Phi(x) \rangle (\iota y.\Psi(y))}{\sigma \mathbf{E}(\iota y.\Psi(y))}$$

Uses of this Existence Rule can always be folded into the following simple derived rules, as our completeness proof will actually show. We have chosen the Existence Rule form we did largely for aesthetic reasons. We leave it to you to check the correctness of the following proposition, which basically amounts to unwinding the definition of **E**.

Proposition 20.7.5 (Derived Existence Rules, *CN*) *The following are derived tableau rules, where p is a parameter that is new to the branch.*

$$\frac{\sigma \langle \lambda x.\Phi(x) \rangle (c)}{\sigma p = c^\sigma} \qquad \frac{\sigma \langle \lambda x.\Phi(x) \rangle (f(t_1, \dots, t_n))}{\sigma p = f^\sigma(t_1, \dots, t_n)} \qquad \frac{\sigma \langle \lambda x.\Phi(x) \rangle (\iota y.\Psi(y))}{\sigma p = [\iota y.\Psi(y)]^\sigma}$$

The Reflexivity Rule is extended automatically. If $\sigma \langle \lambda x.\Phi(x) \rangle (\iota y.\Psi(y))$ occurs on a tableau branch, we can add $\sigma p = [\iota y.\Psi(y)]^\sigma$ where p is a new parameter, using the Derived Existence Rule, then add $\sigma (p = p)$ using our earlier version of reflexivity, and finally we can add $\sigma [\iota y.\Psi(y)]^\sigma = [\iota y.\Psi(y)]^\sigma$ using substitutivity of equality.

We are finished with cases that simply extend rules we have seen before. Next we have important new rules that make use of the special structural features of definite descriptions. Note that one of the rules involves three-way branching.

Definition 20.7.6 (Definite Description Rules, Positive, *CN*)

$$\frac{\sigma \langle \lambda x.\Phi(x) \rangle (\iota y.\Psi(y))}{\sigma \Psi([\iota y.\Psi(y)]^\sigma)} \qquad \frac{\sigma \langle \lambda x.\Phi(x) \rangle (\iota y.\Psi(y))}{\sigma \neg \Psi(p) \mid \sigma \langle \lambda x.x = p \rangle (\iota y.\Psi(y))} \text{ for any parameter } p$$

Definition 20.7.7 (Definite Description Rule, Negative, CN)

$$\frac{\sigma \neg \langle \lambda x. \Phi(x) \rangle (\iota y. \Psi(y))}{\sigma \neg \Psi(p) \mid \sigma \Psi(q) \mid \sigma \langle \lambda x. x = p \rangle (\iota y. \Psi(y))}$$

$$\sigma \neg (p = q)$$

For any parameter p , and new parameter q .

This completes the presentation of our tableau system. Several examples of it in use are in Sect. 20.8, soundness is discussed in Sect. 20.9, and completeness in Sect. 20.10. But before continuing, we discuss some of the intuitions behind the Positive and the Negative Definite Description Rules.

For the second of the Positive rules, if $\sigma \langle \lambda x. \Phi(x) \rangle (\iota y. \Psi(y))$ occurs on a tableau branch, informally we think of $\langle \lambda x. \Phi(x) \rangle (\iota y. \Psi(y))$ as being true at possible world σ . Then in particular, $\iota y. \Psi(y)$ must designate at σ , so if something makes $\Psi(y)$ true at σ , that something must be what $\iota y. \Psi(y)$ designates at σ . That is, if p is a parameter and $\Psi(p)$ holds at world σ then $\langle \lambda x. x = p \rangle (\iota y. \Psi(y))$ must hold. In effect, this implication is what the two parts of the consequence of the rule say when taken together. (Think of an implication as the disjunction of its negated antecedent with its consequent, and tableau branching as disjunction.)

The Negative rule is more complex since it has a three way split instead of two way. If $\sigma \neg \langle \lambda x. \Phi(x) \rangle (\iota y. \Psi(y))$ occurs on a branch, informally $\langle \lambda x. \Phi(x) \rangle (\iota y. \Psi(y))$ is false at possible world σ . It could be false because $\iota y. \Psi(y)$ does not designate at σ , or because it does designate but what is designated does not make $\Phi(x)$ true. Let p be any parameter. If p is not something that $\iota y. \Psi(y)$ designates at σ , then either this is because p does not make $\Psi(y)$ true (represented by the left hand branch), or because p and something else both make $\Psi(y)$ true (represented by the middle branch). If, somehow, we have eliminated both of these, then $\iota y. \Psi(y)$ must designate at σ , and p must be what it designates (this is represented by the right branch). Note that on the right branch, $\iota y. \Psi(y)$ is positively generated, so using the second Predicate Abstraction Rule from Definition 20.7.3 we can get that it designates but does not make $\Phi(x)$ true at σ .

The informal comments we just made reoccur formally as part rule justification in our soundness proof for the tableau system, presented in Sect. 20.9.

We conclude this section by showing that we avoid an apparent (but not real) problem arising from the formulation of our tableau rules.

When talking about definite descriptions we might informally say “the Ψ ,” but subsequently we might find a need to be more formal. Intuitively it doesn’t matter whether we say “the x such that $\Psi(x)$ ” or “the y such that $\Psi(y)$ ”, provided the two variables are substitutable for each other in Ψ . We gave a definition of *substitutable* earlier, Definition 8.1.6, but it is not quite sufficient now since it only refers to accidental binding of a free variable by a quantifier, but now we have the operators λ and ι that can also bind variables. Consequently the earlier definition should be replaced with the following.

Definition 20.7.8 (Substitutability) This replaces Definition 8.1.6. We say a free variable z is *substitutable* for x in $\Phi(x)$ provided no free occurrence of x in $\Phi(x)$ is in a subformula beginning with $(\forall z)$ or $(\exists z)$, or in a subformula beginning with λz , or in a subterm beginning with ιz .

Of course we can *always* replace x by z in a formula; the point of the definition is to specify under what circumstances this doesn't change the intended meaning of the formula. Being substitutable can also be characterized using parameters, which can't be bound at all, and this is probably more intuitive for applications to tableaus. The definition we just gave is equivalent to the following: z is substitutable for x in $\Phi(x)$ provided, for any parameter p that does not occur in the formula, substituting p for x in $\Phi(x)$ and for z in $\Phi(z)$ yields the same result.

Now, here is the (apparent) problem. Suppose we have both of the following on the same tableau branch.

$$\begin{aligned}\sigma \quad & \langle \lambda x.A(x) \rangle (\iota y.\Phi(y)) \quad 1. \\ \sigma \quad & \neg \langle \lambda w.B(w) \rangle (\iota y.\Phi(y)) \quad 2.\end{aligned}$$

We can apply the first of the Predicate Abstraction Rules in Definition 20.7.3 to 1, and we can apply the second to 2 because by Definition 20.7.2 3a, $[\iota y.\Phi(y)]^\sigma$ is positively generated on the branch.

But now suppose that, instead of 1 and 2, we have the following on a branch, where y and z are different but z is substitutable for y in $\Phi(y)$.

$$\begin{aligned}\sigma \quad & \langle \lambda x.A(x) \rangle (\iota y.\Phi(y)) \quad 3. \\ \sigma \quad & \neg \langle \lambda w.B(w) \rangle (\iota z.\Phi(z)) \quad 4.\end{aligned}$$

Because of 3, $[\iota y.\Psi(y)]^\sigma$ is positively generated on the branch, but this is not enough to apply the negative part of Definition 20.7.3 to 4 since that line has $\iota z.\Phi(z)$, and while this is intuitively the same as $\iota y.\Phi(y)$, they are not syntactically the same.

Fortunately this is not actually a problem, because we can show our tableau system has the following derived rule.

Proposition 20.7.9 (Derived Tableau Rule, CN) *Suppose both*

$$\begin{aligned}\sigma \quad & \langle \lambda x.A(x) \rangle (\iota y.\Phi(y)) \\ \sigma \quad & \neg \langle \lambda w.B(w) \rangle (\iota z.\Phi(z))\end{aligned}$$

occur on a tableau branch, where z is substitutable for y . Then in addition to $[\iota y.\Phi(y)]^\sigma$, the item $[\iota z.\Phi(z)]^\sigma$ can be assumed to be positively generated on the branch, and also $[\iota y.\Phi(y)]^\sigma = [\iota z.\Phi(z)]^\sigma$ can be added to the branch end.

Proof Assume we have a tableau with a branch containing the two lines shown. In Fig. 20.2 we show how that branch can be continued to produce a branch meeting the desired conditions. Here are the justifications for the steps that are made.

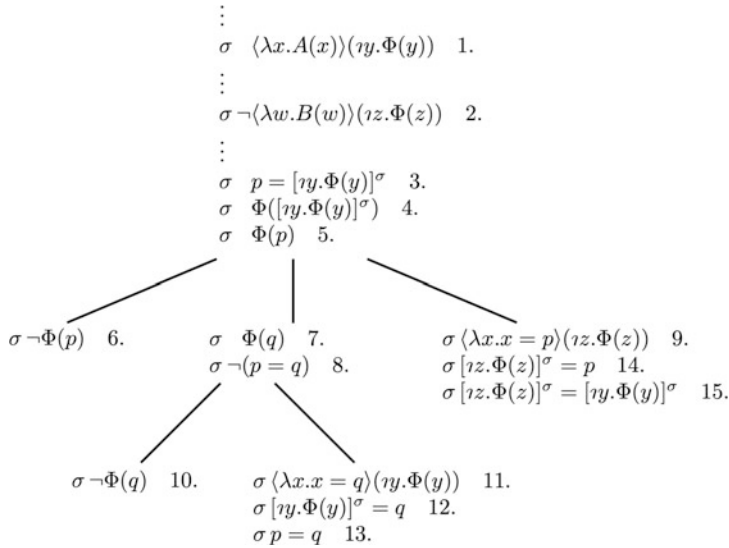


Fig. 20.2 Tableau for Proposition 20.7.9

By assumption, we have a tableau branch containing 1 and 2. Then for a new parameter p , 3 follows from 1 by the Derived Existence Rule, Definition 20.7.5; 4 follows from 1 by the first Positive Definite Description Rule, Definition 20.7.6; 5 follows from 3 and 4 by substitutivity of equality. Next for a new parameter q , 6, 7, 8, and 9 follow from 2 by the Negative Definite Description Rule, Definition 20.7.7; 10 and 11 are from 1 by the second Positive Definite Description Rule, Definition 20.7.6; 12 is from 11 by the first of the Predicate Abstraction Rules, Definition 20.7.3, and 13 is from 3 and 12 by substitutivity of equality. Finally, 14 is from 9 by the first Predicate Abstraction Rule, Definition 20.7.3, and 15 is from 3 and 14 by substitutivity of equality.

Now the original branch has gone through a four-way split. The leftmost branch is closed by 5 and 6; the next to the right is closed by 7 and 10; and the third is closed by 8 and 13. This leaves the rightmost branch as our continuation of the original branch. Note that on it, $[\iota z. \Phi(z)]^\sigma$ is positively generated because of 9, and line 15 is present as promised.

20.7.1 On Origins

We conclude this section with some information on the origins of the rules above. In the first edition of this book, Fitting and Mendelsohn (1998), a tableau system for definite descriptions was given. It was not very satisfactory. One of the features of a good tableau system is that in some appropriate sense each rule application should

replace a formula by one or more simpler formulas. Some of the definite description tableau machinery in the first edition of this book, instead of doing this, made things more complicated. This was not the case for all of the machinery, but some of it allowed certain formulas to be introduced into a tableau as global assumptions, and these were far from simple. Recently two papers, Indrzejczak (2020) and Orlandelli (2021) have reformulated things, one using hybrid logic and the other using labeled deductive systems. We do not discuss their machinery here, but we note that they provided rules for definite descriptions that had in part evolved out of the unfortunate parts of our presentation, but that had a much nicer formal structure. In turn we adapted and reformulated their rules, following the general ideas of the prefixed tableau systems used here, and it is these reformulated rules that have been presented above.

Exercises

Exercise 20.7.1 Prove Proposition 20.7.5.

Exercise 20.7.2 Show the following is a derived rule of our tableau system. The proof of Proposition 20.7.9 can serve as a partial guide.

$$\frac{\sigma \langle \lambda x.A(x) \rangle (\imath y.\Phi(y)) \quad \sigma \langle \lambda w.B(w) \rangle (\imath z.\Phi(z))}{\sigma [\imath y.\Phi(y)]^\sigma = [\imath z.\Phi(z)]^\sigma}$$

20.8 Tableau Examples

Example 20.8.1 We give a *CN* tableau proof of the following, in Fig. 20.3.

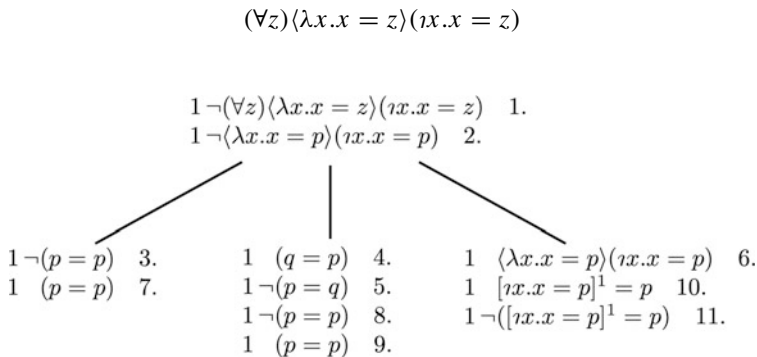


Fig. 20.3 Proof for Example 20.8.1

Note that $\langle \lambda x.x = z \rangle$ is really the property of being z , so the sentence asserts that, for all objects z , the thing that is z has the property of being z . We give the proof for **K**, though as usual the actual choice of modal logic does not really matter.

In the tableau, line 2 is from 1 by an existential rule. Then 3, 4, 5, 6 are from 2 by the negative definite description rule, Definition 20.7.7, where $\Psi(x)$ is $x = p$. Line 7 is by reflexivity, and the left branch is closed. Line 8 is from lines 4 and 5 by substitutivity of equality, line 9 is by reflexivity again, and closure is by 8 and 9. Finally 10 is from 6 by the first Predicate Abstraction Rule, Definition 20.7.3 and now since $[\lambda x.x = p]$ is positively generated on the branch, 11 follows from 2. Closure on the right branch is by 10 and 11. Note: in fact the right branch closed at an earlier stage, because of 2 and 6, but this would not have been *atomic* closure.

Example 20.8.2 In Exercise 20.5.2 we gave a tableau proof of $\mathbf{D}(\lambda x.\Diamond P(x)) \supset \Diamond \mathbf{D}(\lambda x.P(x))$ in *CN K*, using the Russell embedding. Now we give a tableau proof using the machinery just introduced. The tableau is displayed in Fig. 20.4, and the justifications for each step are as follows.

Of course 1 is the negation of the formula to be proved, and 2 and 3 follow by a propositional rule. Then 4 is from 2 by the first Positive Definite Description Rule, Definition 20.7.6, while 5 follows from 2 by Proposition 20.7.5. Then 6 is from 4 and 5 using substitutivity of equality. Then 7 is by a possibility rule from 6, and 8 is by a necessity rule from 3.

The three way split, 9, 10, 11, 12 is from 8 by the Negative Definite Description Rule, Definition 20.7.7. The left branch is closed by 7 and 9. On the right branch, 13 is from 12 by a predicate abstraction rule, Definition 20.7.3, first part. Since $[\lambda x.P(x)]^{1.1}$ is positively generated on the branch, 14 follows from 8 by

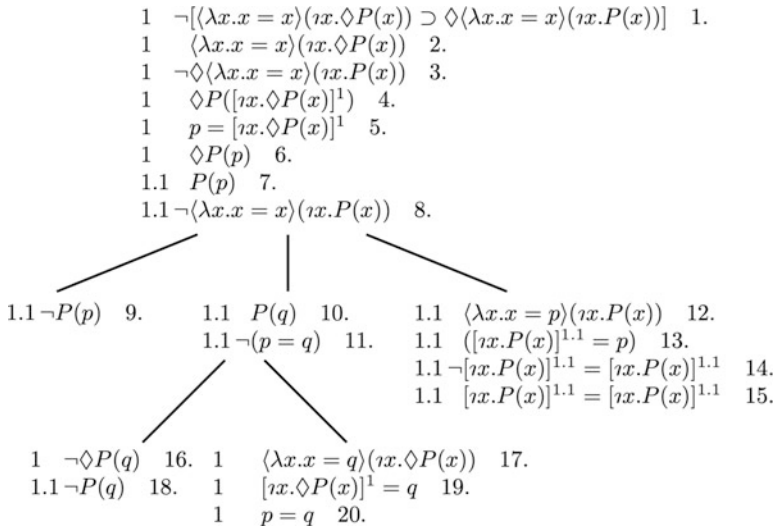


Fig. 20.4 Proof for Example 20.8.2

Definition 20.7.3. Then 15 is by the reflexivity of equality, and right branch closure is by 14 and 15.

Finally, 16 and 17 are from 2 by the second Positive Definite Description Rule, Definition 20.7.6. Now 18 is from 16 by a necessity rule, and this branch is closed using 10 and 18. For the remaining branch, 19 is from 17 using a Predicate Abstraction Rule, Definition 20.7.3, first part, and 20 follows by substitutivity of equality using 19 and 5. Closure is by 11 and 20.

Example 20.8.3 In Example 20.4.4 we discussed a definite description example that involved modality as a temporal operator. Formula (20.14) asserts, under reasonable conditions, that someday the tallest person in the world might not be the tallest person in the world. We gave a direct semantic verification of that formula. We now show it using tableaux. Repeating the formula for convenience, it is the following.

$$[\mathbf{E}(\iota x.T(x)) \wedge (\forall x)\Diamond\neg T(x)] \supset \langle \lambda x.\Diamond\neg\langle \lambda y.x = y \rangle(\iota z.T(z)) \rangle(\iota z.T(z))$$

To make things a little more complicated, we must establish it in a *varying* domain setting, but our current tableau system is constant domain. We can do this by following the advice given in our Strong Recommendation 20.6.2: relativize quantifiers to a formal existence predicate $\mathcal{E}$. But before doing so we note that $\mathbf{E}$ is actually an abbreviation, Definition 15.4.3, and eliminating it introduces a quantifier. Carrying out this elimination we have the following.

$$\begin{aligned} & [\langle \lambda x.(\exists y)(y = x) \rangle(\iota x.T(x)) \wedge (\forall x)\Diamond\neg T(x)] \\ & \supset \langle \lambda x.\Diamond\neg\langle \lambda y.x = y \rangle(\iota z.T(z)) \rangle(\iota z.T(z)) \end{aligned}$$

Then relativizing to an existence predicate, we have the following.

$$\begin{aligned} & [\langle \lambda x.(\exists y)(\mathcal{E}(y) \wedge (y = x)) \rangle(\iota x.T(x)) \wedge (\forall x)(\mathcal{E}(x) \supset \Diamond\neg T(x))] \supset \\ & \langle \lambda x.\Diamond\neg\langle \lambda y.x = y \rangle(\iota z.T(z)) \rangle(\iota z.T(z)) \end{aligned}$$

Figure 20.5 contains a tableau proof of this using the constant domain $\mathbf{K}$ rules. Reasons for the steps are as follows. Lines 2 and 3 are from 1 by negated implication. 4 and 5 are from 2 by conjunction. 6 is from 4 by the first Predicate Abstraction Rule, Definition 20.7.3. 7 and 8 are from 3 and 4 using the Derived Tableau Rule Proposition 20.7.9. 9 is from 6 by existential instantiation, with p being a new parameter. 10 and 11 are from 9 by a conjunction rule. 12 is from 5 by universal instantiation. 13 and 14 are from 12 by an implication rule. 15 is from 14 by a possibility rule, and 16 is from 7 by a necessitation rule, with 17 following by double negation elimination. 18 is from 17 by the first Predicate Abstraction Rule, Definition 20.7.3. 19 is from 17 by the first Positive Definite Description Rule, Definition 20.7.6. 20 is from 19 and 18 by substitutivity of equality, and then 21 follows from this and 11 also by substitutivity of equality. Closure is by 10 and 13, and 15 and 21.

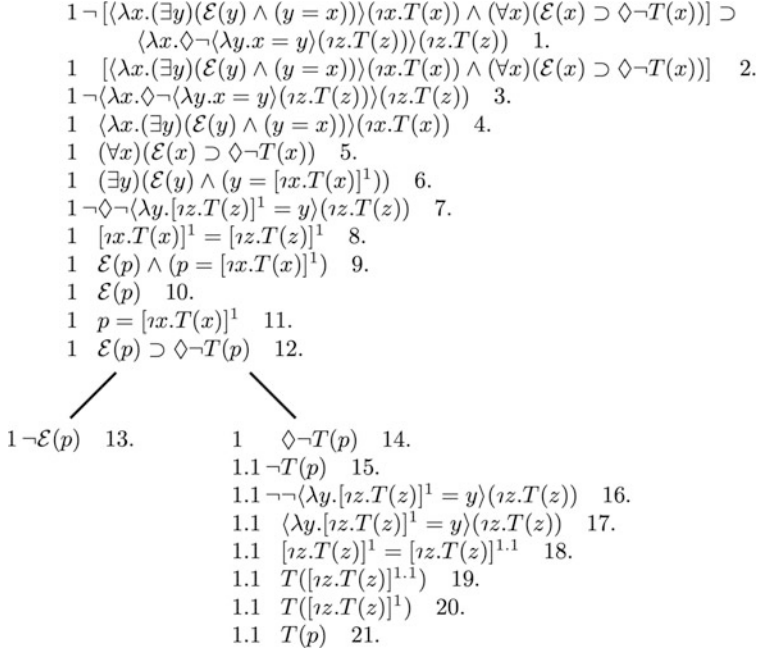


Fig. 20.5 Proof for Example 20.8.3

Exercises

The following exercises ask for tableau proofs. If a constant domain proof is requested, the tableau system from Sect. 20.7 is to be used. If a varying domain proof is needed, relativize the quantifiers of the formula to be proved using an existence predicate, $\mathcal{E}$.

Exercise 20.8.1 Assume that c is a non-rigid constant symbol. Give a tableau proof of the following, $\langle \lambda y. x.y = x \rangle (c, \iota x. \Phi(x)) \supset \langle \lambda x. \Phi(x) \rangle (c)$. It doesn't matter whether the assumption is constant or varying domain. This uses the abbreviation from Definition 14.1.3.

Exercise 20.8.2 Give a constant domain tableau proof of the formula from Exercise 20.4.5:

$$\langle \lambda x. \Box A(x) \rangle (\iota x. \Diamond P(x)) \supset \Diamond \langle \lambda x. A(x) \rangle (\iota x. P(x)).$$

Exercise 20.8.3 Give a constant domain tableau proof of the following, and then a varying domain proof: $(\forall x) \mathbf{E}(\iota y. y = x)$.

Exercise 20.8.4 Give a tableau proof of the following, where it doesn't matter whether we have constant or varying domains, $\mathbf{D}(\iota x. \Phi(x)) \equiv \langle \lambda x. \Phi(x) \rangle (\iota x. \Phi(x))$.

Exercise 20.8.5 Give a varying domain proof of the following, $\mathbf{D}(\iota x.\Phi(x)) \supset [(\exists x)\Phi(x) \supset \mathbf{E}(\iota x.\Phi(x))]$.

Exercise 20.8.6 Give a varying domain proof of the following (see Example 20.4.6).

$$\neg\{\Box\mathbf{E}(\iota z.W(z)) \wedge (\forall w)(\lambda y.\neg\Box(w = y))(\iota z.W(z))\}$$

Exercise 20.8.7 Suppose all P 's are Q 's. Then *the* P and *the* Q will be the same thing, provided both definite descriptions 'make sense'. The following sentence expresses this more precisely.

$$[\mathbf{E}(\iota x.P(x)) \wedge \mathbf{D}(\iota x.Q(x))] \supset \\ [(\forall x)(P(x) \supset Q(x)) \supset \langle \lambda x, y.x = y \rangle(\iota x.P(x), \iota x.Q(x))]$$

Give a varying domain tableau proof of this sentence. Also, what goes wrong if $\mathbf{E}(\iota x.P(x))$ is weakened to $\mathbf{D}(\iota x.P(x))$?

Exercise 20.8.8 The following sentence says that if *the* P designates locally strongly rigidly (Proposition 19.2.7), and *the possible* P designates at all, then the two definite descriptions designate the same thing. Give a tableau proof of this. (It makes no difference whether one assumes varying or constant domains since no quantifiers are involved.)

$$[\langle \lambda x.\Box\langle \lambda y.x = y \rangle(\iota x.P(x)) \rangle(\iota x.P(x)) \wedge \mathbf{D}(\iota x.\Diamond P(x))] \supset \\ \langle \lambda x.\langle \lambda y.x = y \rangle(\iota x.\Diamond P(x)) \rangle(\iota x.P(x))$$

Exercise 20.8.9 Give a tableau proof of the following. (Once again it makes no difference whether it is constant or varying domain.)

$$[\mathbf{D}(\iota z.\Diamond P(z)) \wedge \Box\mathbf{D}(\iota z.P(z))] \supset \langle \lambda x.\Diamond\langle \lambda y.x = y \rangle(\iota z.P(z)) \rangle(\iota z.\Diamond P(z))$$

Exercise 20.8.10 Give a constant domain proof of the following. Then see if a varying domain proof can be given.

$$\Box\mathbf{D}(\iota z.P(z)) \supset (\forall x)[\langle \lambda y.y = x \rangle(\iota z.\Diamond P(z)) \supset \Diamond\langle \lambda y.y = x \rangle(\iota z.P(z))]$$

20.9 Soundness, With Definite Descriptions

When we added predicate abstracts to our formal machinery, in Part V, we wound up with four versions, depending on whether quantifier domains were the same from world to world or could vary, and whether terms always designated or might fail to do so at some possible worlds. Notation for specifying our four versions of a

logic was given in Definition 15.1.1, and tableau systems were provided for all four. Soundness and completeness were proved in detail for only one of these, the *VN* version of **K**, in Chap. 16. This was chosen because it was both the most general and the most complicated. But in Sect. 20.6 we gave *Our Strong Recommendations* that once definite descriptions have been added, then a constant domain version is simplest, most natural, and still allows for the treatment of varying domains by relativizing quantifiers. Earlier we suggested that you try proving soundness and completeness yourselves for versions other than *VN* (without definite descriptions), in Exercises 16.3.1 and 16.8.1. In what follows, in effect, we are assuming you did that for the *CN* version of **K**, and we will just discuss what is needed to provide for definite descriptions. We cover soundness in this section and completeness in Sects. 20.10 and 20.11. We will sketch the necessary details of what we are assuming as need for this comes up.

As we have seen throughout, all tableau soundness proofs have the same general form. Define what it means for a tableau to be satisfiable; show the initial tableau in a tableau construction is satisfiable; show a closed tableau is not satisfiable; and (the largest part of the argument) show each tableau rule application preserves satisfiability. Soundness follows easily from all this, though it might be useful to quickly review Sect. 16.3 before proceeding here. While the proof in Sect. 16.3 was for varying domains and now we have constant domains, there is much similarity between the two. Indeed, constant domains are a simpler structure and so things are substantially easier.

We again make use of what we called a *Context*, Definition 16.1.2, except that now we are assuming the *CN* conditions. A context amounts to the specification of a normal model $\mathcal{M}$ (constant domain now), a valuation v in it, and a prefix function θ mapping prefixes to possible worlds in $\mathcal{M}$. Satisfiability of a tableau in a context is defined as it was in Definition 16.3.2, except that now domains are constant and definite descriptions can appear in formulas. Adapting to these differences should cause little trouble. The main work lies in showing that if a tableau rule is applied to a satisfiable tableau, the result is another satisfiable tableau and, given our previous work, this only needs to be done for the new rules involving definite descriptions, so that is all we discuss here.

Proofs of the preservation of satisfiability when using the definite description rules contained in Definitions 20.7.3 and 20.7.4 are essentially the same as the Predicate Abstraction cases in Sect. 16.3, and we do not repeat these details here. The really new cases are those arising from the rules in Definitions 20.7.6 and 20.7.7. Of these three rules, we only discuss the Negative Definite Description Rule from Definition 20.7.7, the other two being similar and simpler, and consequently good exercises.

We now show that an application of the Negative Definite Description Rule, Definition 20.7.7, preserves tableau satisfiability. We have the following three conditions:

- C1. We have a **K** tableau that has been constructed using the *CN* conditions, and in that tableau there is a branch $\mathcal{B}$ that is satisfiable, in a particular context

that involves the model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$, the valuation v , and the prefix function θ ;

- C2. $\sigma \neg(\lambda x.\Phi(x))(\gamma y.\Psi(y))$ occurs on the branch $\mathcal{B}$;
- C3. The branch $\mathcal{B}$ is extended using the Negative Definite Description Rule. Since the rule in question involves a three-way split, what we must show is that we have branch satisfiability for one of the three branches. Here are the three possibilities in detail, where p is an arbitrary parameter.

Branch 1. $\sigma \neg\Psi(p)$ has been added to $\mathcal{B}$.

Branch 2. $\sigma \Psi(q)$ and $\sigma \neg(p = q)$ have been added to $\mathcal{B}$ where q is a parameter that is new.

Branch 3. $\sigma \langle \lambda x.x = p \rangle(\gamma y.\Psi(y))$ has been added to $\mathcal{B}$.

Here is our argument that one of the three branch versions must be satisfiable. Consider the parameter p ; we have three possibilities concerning its behavior. First, the value assigned to p in our model domain is not something that makes $\Psi(y)$ true in our context; second, the value of p does make $\Psi(y)$ true, but it is not the only thing that does so; third, the value assigned to p is the only thing that makes $\Psi(y)$ true. We treat these three possibilities one by one.

First, suppose $v(p)$ is not an object in the model domain that makes $\Psi(y)$ true at $\theta(\sigma)$. Then $\mathcal{M}, \theta(\sigma) \not\models_v \Psi(p)$, so of course $\mathcal{M}, \theta(\sigma) \models_v \neg\Psi(p)$. Then Branch 1 is satisfiable.

Next, suppose $v(p)$ is an object in the model domain that makes $\Psi(y)$ true at $\theta(\sigma)$, but also there is something else in the domain that makes $\Psi(y)$ true at $\theta(\sigma)$. Let v' be a q -variant of v such that $v'(q)$ is an object in the model domain other than $v(p)$ that makes $\Psi(y)$ true at $\theta(\sigma)$. Using v' instead of v changes no truth value for any prefixed formula on branch $\mathcal{B}$ since q is a new parameter. Then branch $\mathcal{B}$ is satisfiable in the context that is like the original one, but with valuation v' in place of v . But also we have both $\mathcal{M}, \theta(\sigma) \models_{v'} \Psi(q)$ and $\mathcal{M}, \theta(\sigma) \models_{v'} \neg(p = q)$. It follows that Branch 2 is satisfiable, but in an altered context that uses v' instead of v .

Finally, suppose $v(p)$ is an object in the model domain making $\Psi(y)$ true at $\theta(\sigma)$, while nothing else in the domain does so. Then by Definition 20.3.1 $\gamma y.\Psi(y)$ designates at $\theta(\sigma)$, and $v(p)$ is what it designates there. Then $\mathcal{M}, \theta(\sigma) \models_v \sigma \langle \lambda x.x = p \rangle(\gamma y.\Psi(y))$, and so Branch 3 is satisfiable.

We have now showed that the Negative Definite Description Rule preserves tableau satisfiability. The second of the Positive Definite Description Rules, Definition 20.7.6, is similar but a little simpler. If $\sigma \langle \lambda x.\Phi(x) \rangle(\gamma y.\Psi(y))$ occurs on a tableau branch and we know the branch is satisfiable in a particular context, then we know that $\gamma y.\Psi(y)$ designates, so the second of the branch possibilities that came up for the Negative rule has no Positive analog. This accounts for the two way branching in the Positive rule. We leave the details to you, as we also do for the first of the Positive Definite Description Rules.

We have now presented everything that is needed to establish soundness for the CN version of the **K** tableau system allowing definite descriptions.

Exercises

Exercise 20.9.1 Give a full proof that the first of the rules in Definition 20.7.6 preserves tableau satisfiability.

Exercise 20.9.2 Give a full proof that the second of the rules in Definition 20.7.6 preserves tableau satisfiability.

20.10 Hintikka Sets

We have seen Hintikka sets throughout this book, beginning with Sect. 7.5. Each time they reappeared they became more complex, building on what had been the previous version. But always they, and their accompanying version of Hintikka's Lemma, embodied the central part of a completeness proof. This is their last appearance in this book, and the pattern still holds. Our present version, in a language that includes definite descriptions, builds on the version of Hintikka set in Sect. 16.4, specifically on Definition 16.4.1, though that definition is not quite what we need now since it is for the *VN* version of **K** and here we are working with the *CN* version. But this is an easy modification, and we give it without discussing the details. The main thing added now is conditions for definite descriptions.

Definition 20.10.1 (K Hintikka Set, CN Conditions, with Definite Descriptions)

Our earlier Definition 16.4.1 is modified and expanded as follows. The notion of positive generation for an object term is extended so that also $[\iota y. \Psi]^{\sigma}$ is positively generated in H if $\sigma \langle \lambda x. \Phi \rangle (\iota y. \Psi)$ is present in H for some Φ . The propositional and modal conditions H-1 through H-4 from Definition 16.4.1 remain the same, and are not repeated here. Condition H-5 is modified to show constant domain parameters instead of varying domain ones. Condition H-6 is replaced by what is below. Condition H-7 gains one more case. And new conditions H-8 through H-10 are added. What follows are the modifications and additions. Throughout, p and q are parameters.

H-5 The following quantifier closure conditions are met.

- $$\begin{aligned} \sigma (\exists x) \Psi(x) \in H &\implies \sigma \Psi(p) \in H \text{ for some parameter } p \\ \sigma \neg(\forall x) \Psi(x) \in H &\implies \sigma \neg\Psi(p) \in H \text{ for some parameter } p \\ \sigma (\forall x) \Psi(x) \in H &\implies \sigma \Psi(p) \in H \text{ for every parameter } p \\ \sigma \neg(\exists x) \Psi(x) \in H &\implies \sigma \neg\Psi(p) \in H \text{ for every parameter } p \end{aligned}$$

H-6 The following Equality conditions are met for every parameter p , pseudo-closed object terms t, u , and atomic formula $\Phi(x)$:

$$\begin{aligned} \sigma(p = p) &\in H, \text{ where } \sigma \text{ occurs in } H \\ \sigma(t = u) &\in H \text{ and } \tau\Phi(t) \in H \implies \tau\Phi(u) \in H \end{aligned}$$

H-7 The following Predicate Abstraction conditions are met. For any constant symbol c , n -place function symbol f , pseudo-closed object terms $t_1, \dots, t_n$, and definite description $\iota y.\Psi$:

$$\begin{aligned} \sigma\langle\lambda x.\Phi(x)\rangle(c) &\in H \implies \sigma\Phi(c^\sigma) \in H \\ \left. \begin{array}{l} \sigma\neg\langle\lambda x.\Phi(x)\rangle(c) \in H \\ \text{and} \\ c^\sigma \text{ is positively generated in } H \end{array} \right\} &\implies \sigma\neg\Phi(c^\sigma) \in H \\ \sigma\langle\lambda x.\Phi(x)\rangle(f(t_1, \dots, t_n)) &\in H \implies \sigma\Phi(f^\sigma(t_1, \dots, t_n)) \in H \\ \left. \begin{array}{l} \sigma\neg\langle\lambda x.\Phi(x)\rangle(f(t_1, \dots, t_n)) \in H \\ \text{and} \\ f^\sigma(t_1, \dots, t_n) \text{ is positively generated in } H \end{array} \right\} &\implies \sigma\neg\Phi(f^\sigma(t_1, \dots, t_n)) \in H \\ \sigma\langle\lambda x.\Phi(x)\rangle(\iota y.\Psi) &\in H \implies \sigma\Phi([\iota y.\Psi]^\sigma) \in H \\ \left. \begin{array}{l} \sigma\neg\langle\lambda x.\Phi(x)\rangle(\iota y.\Psi) \in H \\ \text{and} \\ [\iota y.\Psi]^\sigma \text{ is positively generated in } H \end{array} \right\} &\implies \sigma\neg\Phi([\iota y.\Psi]^\sigma) \in H \end{aligned}$$

H-8 The following Existence conditions are met:

$$\begin{aligned} \sigma\langle\lambda x.\Phi(x)\rangle(c) &\in H \implies \sigma(p = c^\sigma) \in H \text{ for some } p \\ \sigma\langle\lambda x.\Phi(x)\rangle(f(t_1, \dots, t_n)) &\in H \implies \sigma(p = f^\sigma(t_1, \dots, t_n)) \in H \\ &\text{for some } p \\ \sigma\langle\lambda x.\Phi(x)\rangle(\iota y.\Psi(y)) &\in H \implies \sigma(p = [\iota y.\Psi(y)]^\sigma) \in H \\ &\text{for some } p \end{aligned}$$

H-9 The following Positive Definite Description conditions are met:

$$\begin{aligned} \sigma\langle\lambda x.\Phi(x)\rangle(\iota y.\Psi(y)) &\implies \sigma\Psi([\iota y.\Psi(y)]^\sigma) \\ \sigma\langle\lambda x.\Phi(x)\rangle(\iota y.\Psi(y)) &\implies \left\{ \begin{array}{l} \sigma\neg\Psi(p) \in H \\ \text{or} \\ \sigma p = [\iota y.\Psi(y)]^\sigma \in H \end{array} \right\} \text{ for each } p \end{aligned}$$

H–10 The following Negative Definite Description condition is met:

$$\sigma \neg \langle \lambda x. \Phi(x) \rangle (\iota y. \Psi(y)) \implies \left\{ \begin{array}{l} \sigma \neg \Psi(p) \in H \\ \text{or} \\ \sigma \Psi(q), \sigma \neg(p = q) \in H \\ \quad \text{for some } q \\ \text{or} \\ \sigma \langle \lambda x. x = p \rangle (\iota y. \Psi(y)) \in H \end{array} \right\} \text{ for each } p$$

A few comments before we move on. The Existence conditions in H–8 do not match the tableau Existence rules from Definitions 15.4.5 and 20.7.4, but rather the derived rules in Proposition 20.7.5. Thus our completeness proof actually establishes completeness with the Derived Rules in place of the official rules we adopted. Since these are *derived* rules, completeness of our official system is indirectly established.

Of course the big item that is still missing is the appropriate version of Hintikka’s Lemma, that every Hintikka set is satisfiable in a normal model. We devote a section of its own to this.

20.11 Hintikka’s Lemma

The last time we discussed Hintikka sets in detail was in Chap. 16, and that was notably difficult. We deliberately chose to work with the VN conditions as a fully representative example, because that involved every complexity that had come up in this book. Consequently the work of proving an appropriate version of Hintikka’s Lemma was spread broadly, over Sects. 16.4, 16.5, 16.6, and 16.7. Even though we have predicate abstracts to deal with now, things are actually much easier because we don’t have varying domains and that makes a substantial difference.

The proof of Hintikka’s Lemma in Chap. 16 involved varying domains and subscripted parameters. We now have constant domains so of course parameters don’t have subscripts. In the earlier proof it was necessary to construct a witness world, because we needed a place where the designations of object terms existed for those object terms about which we had insufficient information. This complication vanishes now because in constant domain models all objects exist everywhere, and nothing special needs to be done. We assume you can ignore the now irrelevant parts of our earlier proof. The chief complexity that carries over is the need for equivalence classes. When equality is present, the direct construction of a model $\mathcal{M}$ from a Hintikka set does not produce a normal model, but rather one in which the equality symbol is interpreted by an equivalence relation. That model must be factored, by bringing in equivalence classes, to produce the normal model we denoted earlier by $\mathcal{M}$. The basic details for doing this are in Sect. 11.7. We

assume that you have a general understanding of all this, and we only discuss the complications that definite descriptions directly bring in.

A proof of Hintikka's Lemma always involves a Complete Induction argument on the complexity of members of the Hintikka set. Complexity is measured here by *degree*, and Definition 14.1.2 gets the obvious updating.

Definition 20.11.1 (Degree) The degree of a first-order modal formula allowing predicate abstraction and definite descriptions is the total number of occurrences of $\neg$, $\wedge$, $\vee$, $\supset$, $\square$, $\diamond$, $\forall$, $\exists$, λ , and ι in it. Symbols occurring within definite descriptions are counted, except that symbols occurring within superscripted definite descriptions are *not* counted.

As expected, the Hintikka set conditions, except for H-1 and H-6, all say that if some prefixed formula is present, so are one or more prefixed formulas of lower degree. We leave it to you to check this.

Hintikka's Lemma has to do with satisfiability for tableaux, and so we now need to extend Definition 16.1.3, for Object Term Evaluation, to include definite descriptions. (There is also a minor change to our earlier definition: since we are working with constant domains now, parameters don't have subscripts.) One more category is added to the earlier table in Definition 16.1.3, making use of Definition 20.3.1, as follows.

Definition 20.11.2 (Addition to Definition 16.1.3)

Category	Definition
Definite Description	$\llbracket (\iota x. \Psi(x))^\sigma \rrbracket = \begin{cases} \mathcal{J}(\iota x. \Psi(x), \theta(\sigma)) & \\ \text{if } \mathcal{J}(\iota x. \Psi(x), \theta(\sigma)) & \\ \text{is defined} & \\ \text{undefined} & \text{otherwise} \end{cases}$

Finally, we are ready for the main event. (We note without proof that Proposition 16.2.1 extends to allow definite descriptions.)

Proposition 20.11.3 (Hintikka's Lemma) *Let H be a Hintikka Set according to Definition 20.10.1, for the logic $\mathbf{K}$ with the CN conditions. Then H is satisfiable in a normal $\mathbf{K}$ model meeting the CN conditions.*

It would help if you looked at the second part of the proof of Proposition 11.7.2, where a model $\mathcal{M}$, in which “=” is interpreted by an equivalence relation, is converted into $\overline{\mathcal{M}}$, where it is interpreted by equality. Exactly the same construction applies now, and we do not repeat all the details. We just sketch the basics, and verify that definite descriptions behave properly in our model.

We are, however, faced with a problem we did not have before. In our earlier proofs of Hintikka's Lemma for quantified modal logics we used information from a Hintikka set to define a skeleton, and an interpretation in it. That interpretation determined the value of all formulas, in the skeleton, giving us a model. But definite descriptions bring problems that we have mentioned before. To say how

a definite description $\iota y.\Psi$ behaves we need to know how the formula Ψ behaves. But formulas can contain definite descriptions, so to know how formulas behave, we need to know how definite descriptions behave. We are in a circular situation. Here is how we will deal with it.

In a model, a definite description $\iota y.\Psi$ determines some value at each world where it designates. The machinery that determines whether a definite description designates, and if so what, is reflected in the Hintikka set conditions H-9 and H-10. If it does designate, the final two conditions of H-7 tell us what we can do with the thing it designates. Well we will simply base the definition of our interpretation for definite descriptions entirely on H-7, and ignore the two problematic conditions. That is enough to define a model without running into the circularity issues we mentioned above. We will show that, in the resulting model, all members of the Hintikka set are satisfied. And as a *consequence* of this, definite descriptions actually meet the appropriate conditions as to when and what they designate.

Proof Let H be a Hintikka set according to Definition 20.10.1. We first construct a model $\mathcal{M} = \langle \mathcal{G}, \mathcal{R}, \mathcal{D}, \mathcal{I} \rangle$ as follows.

1. $\mathcal{G}$ is the set of all prefixes in H .
2. Accessibility (recall we are using **K**) is given by $\sigma \mathcal{R} \sigma.n$ whenever $\sigma, \sigma.n \in \mathcal{G}$.
3. The constant domain $\mathcal{D}$ is the set of all parameters.
4. $\mathcal{I}$ is specified as follows.

Relation Symbols in $\mathcal{M}$: For an n -place relation symbol,

$$\mathcal{I}(R, \sigma) = \{ \langle p_1, \dots, p_n \rangle \mid \sigma R(p_1, \dots, p_n) \in H \}$$

for (constant domain) parameters $p_1, \dots, p_n$.

Non-Rigid Constant Symbols in $\mathcal{M}$: For a non-rigid constant symbol, $\mathcal{I}(c, \sigma)$ is defined if $\sigma \langle \lambda x.\Phi \rangle(c) \in H$ for some predicate abstract $\langle \lambda x.\Phi \rangle$. And if it is defined, $\mathcal{I}(c, \sigma)$ is the alphabetically first parameter p such that $\sigma(p = c^\sigma) \in H$.

Non-Rigid Function Symbols in $\mathcal{M}$: For a non-rigid n -place function symbol, $\mathcal{I}(f, \sigma)$ is defined as a partial function at possible world σ with $\langle p_1, \dots, p_n \rangle$ in its domain (for $p_i \in \mathcal{D}$) if $\sigma \langle \lambda x.\Phi \rangle(f(p_1, \dots, p_n)) \in H$ for some predicate abstract $\langle \lambda x.\Phi \rangle$. And if it is defined, $\mathcal{I}(f, \sigma)(p_1, \dots, p_n)$ is the alphabetically first parameter p such that $\sigma p = f^\sigma(p_1, \dots, p_n) \in H$.

Definite Descriptions in $\mathcal{M}$: For a definite description, $\mathcal{I}(\iota y.\Psi, \sigma)$ is defined if $\sigma \langle \lambda x.\Phi \rangle(\iota y.\Psi(y)) \in H$ for some predicate abstract $\langle \lambda x.\Phi \rangle$. And if it is defined, $\mathcal{I}(\iota y.\Psi, \sigma)$ is the alphabetically first parameter p such that $\sigma(p = [\iota y.\Psi]^\sigma) \in H$.

In specifying the interpretation $\mathcal{I}$ we always made use of an alphabetically first parameter meeting a condition. This was simply so that some particular item was specified. But there are two issues. First why must there be such a parameter and, second, why this one instead of some other parameter that H “thinks” is equal to it?

The first question is easy. There must be such a parameter because of condition H-8. As to the second question, let us use the non-rigid constant symbol case as representative. Suppose $\tau(p = q) \in H$, where p and q are parameters, and $\sigma(p = c^\sigma) \in H$. Then we will also have $\sigma(q = c^\sigma) \in H$ using condition H-6. Then up to what H “thinks” is equality, the actual choice of parameter really doesn't matter.

This completes the definition of the model $\mathcal{M}$. Next we convert it into a model $\overline{\mathcal{M}}$ that is normal.

A relation $\sim$ is defined on $\mathcal{D}$ by setting $p \sim q$ if $\sigma p = q \in H$ for some (equivalently, for any) prefix σ . This is an equivalence relation on $\mathcal{D}$ (see Sects. 11.1 and 11.7), and so partitions $\mathcal{D}$ into disjoint equivalence classes. Denote the equivalence class containing the parameter p by $\overline{p}$.

We next form a new model $\overline{\mathcal{M}} = \langle \mathcal{G}, \mathcal{R}, \overline{\mathcal{D}}, \overline{\mathcal{I}} \rangle$ from the old model $\mathcal{M}$. The set of possible worlds and the accessibility relation are the same as in $\mathcal{M}$. The domain $\overline{\mathcal{D}}$ is the set of all equivalence classes, $\{\overline{p} \mid p \in \mathcal{D}\}$. And finally, the interpretation function, $\overline{\mathcal{I}}$, is defined as follows.

Relation Symbols in $\overline{\mathcal{M}}$: For an n -place relation symbol,

$$\langle \overline{p_1}, \dots, \overline{p_n} \rangle \in \overline{\mathcal{I}}(R, \sigma) \text{ if } \langle p_1, \dots, p_n \rangle \in \mathcal{I}(R, \sigma).$$

Non-Rigid Constant Symbols in $\overline{\mathcal{M}}$: For a non-rigid constant symbol, $\overline{\mathcal{I}}(c, \sigma)$ is defined if $\mathcal{I}(c, \sigma)$ is defined, and has the value $\overline{\mathcal{I}(c, \sigma)}$.

Non-Rigid Function Symbols in $\overline{\mathcal{M}}$: For a non-rigid n place function symbol, $\overline{\mathcal{I}}(f, \sigma)$ is defined if $\mathcal{I}(f, \sigma)$ is defined, $\langle \overline{p_1}, \dots, \overline{p_n} \rangle$ is in the domain of $\overline{\mathcal{I}}(f, \sigma)$ if $\langle p_1, \dots, p_n \rangle$ is in the domain of $\mathcal{I}(f, \sigma)$, and the resulting function value on equivalence classes is given by $\overline{\mathcal{I}}(f, \sigma)(\overline{p_1}, \dots, \overline{p_n}) = \overline{\mathcal{I}(f, \sigma)(p_1, \dots, p_n)}$.

Definite Descriptions in $\overline{\mathcal{M}}$: For a definite description, $\overline{\mathcal{I}}(\iota y. \Psi(y), \sigma)$ is defined if $\mathcal{I}(\iota y. \Psi(y), \sigma)$ is defined, and has the value $\overline{\mathcal{I}(\iota y. \Psi(y), \sigma)}$.

We leave it to you to verify that the conditions above do constitute a proper definition of a model $\overline{\mathcal{M}}$, and that model is normal. Adapt the ideas from Sect. 11.7.

The goal of the proof of Hintikka's Lemma is to show that for each prefixed formula $\sigma \Phi \in H$, we have that Φ is true in $\overline{\mathcal{M}}$ at possible world σ using the valuation v_0 that maps each parameter p to the corresponding equivalence class $\overline{p}$ in $\overline{\mathcal{D}}$. The proof is by induction on the complexity (degree) of Φ , where the induction hypothesis is that the condition is known to hold for each prefixed formula $\tau \Psi$, where Ψ is simpler than Φ . The proof is by cases, depending on the form of formula Φ . In Sect. 11.7 we have already covered all the cases that do not involve definite descriptions. It is only the definite description cases that we give here.

Positive Definite Description Suppose $\sigma \langle \lambda x. \Phi(x) \rangle (\iota y. \Psi(y)) \in H$, and assume the induction hypothesis is known for prefixed formulas in H involving formulas that are simpler than $\langle \lambda x. \Phi(x) \rangle (\iota y. \Psi(y))$. We must show that in the model $\overline{\mathcal{M}}$, using valuation v_0 , exactly one member of the domain makes $\Psi(y)$ true at σ , and

that member also makes $\Phi(x)$ true at σ . Then we can conclude that $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \langle \lambda x. \Phi(x) \rangle (\iota y. \Psi(y))$.

Since H is a Hintikka set, we have the following.

By H-7: $\sigma \Phi([\iota y. \Psi(y)]^\sigma) \in H$

By H-8: $\sigma p_0 = [\iota y. \Psi(y)]^\sigma \in H$ for some parameter p_0

By H-9: $\sigma \Psi([\iota y. \Psi(y)]^\sigma) \in H$

By H-9: $\sigma \neg \Psi(p) \in H$ or $\sigma p = [\iota y. \Psi(y)]^\sigma \in H$ for each parameter p

Since all of the formulas above are simpler than $\langle \lambda x. \Phi(x) \rangle (\iota y. \Psi(y))$, by the induction hypothesis we have the following, in which we take $[\iota y. \Psi(y)]^\sigma$ to have the value $\overline{\mathcal{I}}(\iota y. \Psi(y), \sigma)$, where $\overline{\mathcal{I}}$ is as defined from $\mathcal{I}$, and both are defined earlier in this section.

$$\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \Phi([\iota y. \Psi(y)]^\sigma) \quad (20.22)$$

$$\overline{\mathcal{M}}, \sigma \Vdash_{v_0} p_0 = [\iota y. \Psi(y)]^\sigma \quad (20.23)$$

$$\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \Psi([\iota y. \Psi(y)]^\sigma) \quad (20.24)$$

$$\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \neg \Psi(p) \text{ or } \overline{\mathcal{M}}, \sigma \Vdash_{v_0} p = [\iota y. \Psi(y)]^\sigma \text{ for each } p \quad (20.25)$$

Since $\overline{\mathcal{M}}$ is a normal model the equality symbol is interpreted by the equality relation, so condition (20.23) tells us that the domain value assigned to $[\iota y. \Psi(y)]^\sigma$ is $v_0(p_0)$, that is, $\overline{p_0}$. Then by (20.24) we have $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \Psi(p_0)$, so the domain value $\overline{p_0}$ makes $\Psi(y)$ true at σ .

Now suppose that $\overline{p}$ is a member of the domain $\overline{\mathcal{D}}$ of $\overline{\mathcal{M}}$ that is different from $\overline{p_0}$. That is, $v_0(p)$ and $v_0(p_0)$ are different. Once again since $\overline{\mathcal{M}}$ is a normal model, $\overline{\mathcal{M}}, \sigma \nVdash_{v_0} p = p_0$ and then using (20.23), $\overline{\mathcal{M}}, \sigma \nVdash_{v_0} p = [\iota y. \Psi(y)]^\sigma$. It follows by (20.25) that $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \neg \Psi(p)$, and so $\overline{\mathcal{M}}, \sigma \nVdash_{v_0} \Psi(p)$. Thus $v_0(p)$, or $\overline{p}$, is not a domain value making $\Psi(y)$ true at σ .

We have shown that exactly one thing in $\overline{\mathcal{D}}$ makes $\Psi(y)$ true at σ , namely $\overline{p_0}$. It thus meets the conditions needed to properly interpret $\iota y. \Psi(y)$ at σ .

Finally, (20.22) and (20.23) combined tell us that $\overline{p_0}$ makes $\Phi(x)$ true at σ . We now have everything we needed to conclude that $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \langle \lambda x. \Phi(x) \rangle (\iota y. \Psi(y))$.

Negative Definite Description Suppose $\sigma \neg \langle \lambda x. \Phi(x) \rangle (\iota y. \Psi(y)) \in H$, and we assume the induction hypothesis holds for simpler formulas, We must show that $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \neg \langle \lambda x. \Phi(x) \rangle (\iota y. \Psi(y))$. Hintikka condition H-10 applies and for each parameter there are three possibilities, each of which involves a formula of lower

degree than that of $\neg(\lambda x.\Phi(x))(\iota y.\Psi(y))$ and thus is a formula to which the induction hypothesis applies.

For each parameter p :

$$\sigma \neg\Psi(p) \in H, \text{ or} \quad (20.26)$$

$$\sigma \Psi(q), \sigma \neg(p = q) \in H \text{ for some parameter } q, \text{ or} \quad (20.27)$$

$$\sigma \langle \lambda x.x = p \rangle (\iota y.\Psi(y)) \in H \quad (20.28)$$

The proof divides into cases that are based on features of the model $\overline{\mathcal{M}}$, and not on the Hintikka set H itself.

Case 1 $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \neg\Psi(p)$ for every parameter p .

Case 2 $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \Psi(p_0)$ for some parameter p_0 .

Subcase 2a $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \Psi(q)$ for some parameter q , where $\overline{p_0} \neq \overline{q}$.

Subcase 2b For every parameter q such that $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \Psi(q)$, we have $\overline{p_0} = \overline{q}$.

Suppose we are in Case 1. In this case $\Psi(y)$ fails at possible world σ for every member $v_0(p) = \overline{p}$ in the domain $\overline{\mathcal{D}}$, and so $\iota y.\Psi(y)$ does not designate at possible world σ since $\Psi(y)$ is not true of anything. Since no predicate abstract is true of a non-designating term, we have that $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \neg(\lambda x.\Phi(x))(\iota y.\Psi(y))$, which is what we want.

For the rest of this proof, assume we are not in Case 1, and so we are in Case 2 and $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \Psi(p_0)$ for a particular parameter p_0 , fixed for the rest of the proof. By our induction hypothesis, if we had $\sigma \neg\Psi(p_0) \in H$ we would have $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \neg\Psi(p_0)$ and we do not, and so we do not have (20.26) with p being p_0 . Thus from here on we are in one of Subcase 2a or Subcase 2b, and we have one of (20.27) or (20.28), with p being p_0 .

Assume next that we are in Subcase 2a, that is, we have $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \Psi(q)$ and $\overline{p_0} \neq \overline{q}$. We also have $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \Psi(p_0)$. Then once again $\iota y.\Psi(y)$ does not designate at possible world σ , this time because $\Psi(y)$ is true of two distinct things at σ . And then again we have $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \neg(\lambda x.\Phi(x))(\iota y.\Psi(y))$ because all predicate abstracts are false on non-designating terms.

Finally assume we are in Subcase 2b. Since we are in Case 2 there is a parameter p_0 with $\overline{p_0} \in \overline{\mathcal{D}}$ making $\Psi(y)$ true at σ and, since we are in Subcase 2b, this is the only member to do so. Consequently $\overline{p_0}$ has the appropriate semantic properties for being the value of $\iota y.\Psi(y)$ at possible world σ .

We cannot have (20.27) because if we did, our induction hypothesis would give us $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \Psi(q)$ and $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \neg(p_0 = q)$, which would put us in Subcase 2a, while we are in Subcase 2b. But then we must have (20.28) and so $\sigma \langle \lambda x.x = p_0 \rangle (\iota y.\Psi(y))$ is in H and, by H-7, so is $\sigma ([\iota y.\Psi(y)]^\sigma = p_0)$. Then by our induction hypothesis, we have $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} ([\iota y.\Psi(y)]^\sigma = p_0)$.

Since $\sigma \langle \lambda x.x = p_0 \rangle (\imath y.\Psi(y)) \in H$, then $[\imath y.\Psi(y)]^\sigma$ is positively generated in H and, since H is a Hintikka set and $\sigma \neg \langle \lambda x.\Phi(x) \rangle (\imath y.\Psi(y)) \in H$, then $\sigma \neg \Phi([\imath y.\Psi(y)]^\sigma) \in H$ by H-7, and hence also $\sigma \neg \Phi(p_0) \in H$, and so $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \neg \Phi(p_0)$, once again by our induction hypothesis.

We now have $\overline{\mathcal{M}}, \sigma \Vdash_{v_0} \neg \langle \lambda x.\Phi(x) \rangle (\imath y.\Psi(y))$ in this final case too. To summarize the reasons: We know that $\imath y.\Psi(y)$ designates, with value $\overline{p_0}$, at σ because this is the one and only thing making $\Psi(y)$ true at σ . And we know that $v(p_0) = \overline{p_0}$ makes $\Phi(x)$ false at σ .

This ends the proof.

20.12 Completeness

As we have seen several times in this book, but in simpler settings, Hintikka sets arise from attempted tableau proof constructions. One needs to formulate a systematic tableau construction algorithm, based on the *CN* rules for **K** but incorporating the predicate abstraction rules from Sect. 20.7. This algorithm should be a *fair* one, so that every applicable rule is eventually applied. A tableau constructed following such an algorithm either will lead to a proof, or it will not. If it does not, it will generate a tableau with at least one open branch, probably an infinite one. The set of formulas on such a branch will be a Hintikka set according to the definition above. The formulation of such a fair algorithm is something that we leave to you. We assume it can be done, indeed, that it has been done, and we continue under that assumption (Of course an appropriately tailored maximal consistent set construction could also be used, if you prefer.).

Finally, as usual, suppose we have a formula Φ and we begin a tableau with $1 \neg \Phi$. Then if we follow a fair proof algorithm for tableau construction, either we will produce a proof, or there must be a Hintikka set containing $1 \neg \Phi$. In the latter case, by Hintikka's Lemma, Proposition 20.11.3, there must be a normal model in which $\neg \Phi$ is true at a possible world, and so Φ is not valid. We thus have the following.

Theorem 20.12.1 (Completeness) *Let Φ be a sentence in our quantified modal language allowing equality and definite descriptions, and assume we are using the CN tableau rules for **K**, the rules for equality, and the definite description rules given earlier in this chapter. If Φ is not provable then there is a normal **K** model invalidating Φ . Conversely, if Φ is valid in all normal models, it has a tableau proof.*

References

- Blamey, S. (1986). Partial logic. In D. M. Gabbay, & F. Guentner (Eds.), *Handbook of philosophical logic* (Chap. III–1, pp. 1–70). 1983–1989.
- Fitting, M. & Mendelsohn, R. (1998). *First-Order Modal Logic*. Paperback, 1999. Errata at <http://melvinfitting.org/errata/errata.html>. Kluwer.
- Frege, G. (1893). *Grundgesetze der Arithmetik*. Pohle. Reprinted in 1962 Olms: Hildesheim; partial translation in Furth (1967).
- Furth, M. (Ed.). (1967). *The basic laws of arithmetic: Exposition of the system*. University of California Press. Furth, Montgomery (tr.).
- Gabbay, D. M. & Guentner, F. (Eds.). (1983–1989). *Handbook of philosophical logic* (Four volumes). Kluwer.
- Indrzejczak, A. (2020). Existence, definedness and definite descriptions in hybrid modal logic. In N. Olivetti, R. Verbrugge, S. Negri, & G. Sandu (Eds.), *Advances in modal logic* (Vol. 13, pp. 349–368). College Publications.
- Kripke, S. (2011). Russell's notion of scope. *Philosophical troubles*. (Vol. 1, 8, pp. 225–253). Oxford University Press. Collected Papers.
- Orlandelli, E. (2021). Labelled calculi for quantified modal logics with definite descriptions. *Journal of Logic and Computation*, 31(3), 923–946.
- Quine, W. V. O. (1961). *Reference and modality* (pp. 139–159). Harvard University Press.
- Rodriguez-Consuegra, F. A. (2000). Frege, Peano and Russell on descriptions: A comparison. *Russell: The Journal of Bertrand Russell Studies*, 20, 5–25.
- Russell, B. (1905). On denoting. *Mind*, 14, 479–493. Reprinted in Robert C. Marsh, ed., *Logic and Knowledge: Essays 1901–1950, by Bertrand Russell*, Allen & Unwin, London, 1956.
- Smullyan, A. F. (1948). Modality and description. *The Journal of Symbolic Logic*, 13, 31–37.
- Whitehead, A. N. & Russell, B. (1910–1913). *Principia mathematica* (Three volumes, 1st ed.). Cambridge University Press.
- Whitehead, A. N. & Russell, B. (1925–1927). *Principia mathematica* (Three volumes, 2nd ed.). Cambridge University Press.

Afterword

We are at the end. What final messages do we have?

Throughout this book we have gradually built up more and more complex formal systems, within which modal aspects of informal language can be represented, examined, and experimented with. We have tried to be as broad as possible, within the general framework we adopted. Thus we presented not a modal logic, but a family of them. Different ones are appropriate for different purposes. And there are more out there in the literature besides those we considered—many more indeed. We discussed quantification in both actualist and possibilist versions. We introduced rigid and non-rigid designators—constant and function symbols—and this led up to definite descriptions. For all this, we used tableau systems, which are generally suitable for automation. That is, computer programs that make use of this machinery are possibilities. What remains?

There were a number of places where there were choice points, and we made our choices while ignoring the others. Most notably, we began with classical logic. We could have chosen relevance logic, or intuitionistic logic, or a connexive logic, for instance. Much less is known about modality in these contexts, though such knowledge is growing. This would have been a very different book.

Our tableau systems were of the prefixed variety. We pointed out, in one section, that other styles of tableaux exist and have their (sometimes considerable) virtues. We did not pursue this topic any further here. We did not examine natural deduction formulations, or follow up on axiomatics beyond a certain point.

When it came to constant and function symbols that failed to denote, we made use of predicate abstraction machinery, and simply said we treat such abstracts as being false of non-denoting terms. We could have introduced a partial logic at this point: one of Kleene's three-valued logics, or logic of paradox, or first degree entailment for instance, and developed things in such a context. We did not.

In our treatment of terms that do denote, we arranged things so that denoting meant denoting in some possible world. We could have based things on free logic, and allowed the domain of a model simply to be larger than the union of the various

possible world domains. Then denoting would be a broader notion than denoting at some possible world. We did not take this route.

Our treatment of definite descriptions was modeled on the famous one of Russell. That was designed to keep things two-valued (relative to a possible world, though). One might quite naturally think of a sentence involving a non-denoting definite description as lacking a classical truth value. We did not explore this possibility.

Please do not think that we have made all the *right* choices. We have made the ones that were the most appealing to us, the ones for which formal machinery could be easily and naturally developed, the ones that fit most comfortably with our informal philosophical ideas of how language and its constructs behave. If your ideas and inclinations are different, create and investigate appropriate alternatives to ours. Properly formulated machinery of formal logic does not dictate to its user what is the case; but it does tell you what is the case relative to the choices you made when you set up your formalization. One of the beauties of formalization is that mathematical rigor keeps flights of fancy from getting too fanciful. A kind of loose dogma here is this. Informal logical ideas that can be formalized have been shown to be coherent. They have not been shown to be either true or useful. This is where philosophical argumentation comes in.

Index

A

Abélard, P., 290–292
 Abstraction rule, 338
 Accessibility, 57–59, 61, 79, 86, 92, 99, 132, 137, 148, 149, 153, 154, 174, 179, 190, 191, 210, 220, 245, 251, 275, 303, 319, 320, 323, 325, 371, 387, 397, 398, 403, 415, 442, 443
 Actualist quantification, viii, 173, 179, 268, 386, 413
 Actualists, viii, 173, 183, 186, 192, 253, 267–269, 324, 416, 425
 Actuality operator, 301–305
 Adverbial, 52, 56, 57
 Alethic, 52, 60, 61, 234, 289, 388, 394, 396
 Alternative world, 79, 269, 317
 Always designate, 256, 285, 307, 309–311, 314–320, 326, 331, 332, 335–341, 343–348, 351, 381, 385, 397, 405, 426
 Ammonius, 68
 Analytic proof, 4, 31
 Anti-monotonic frame, 193
 Anti-monotonicity, 190, 219, 220, 268–271, 275
 Anti-Realist, 74
 Apodeictic, 52, 54, 64, 73, 74
 A posteriori, 233, 396
 A priori, 55, 230, 233, 388, 396
 Aquinas, S.T., 290, 292
 Aristotle, 53, 55–56, 63–69, 290–292
 Arity, 159, 312
 Artemov, S.N., 118, 119

Assertoric, 52, 54, 73, 74
 Atomic formula, 4, 6, 39, 41, 46, 135, 138, 140, 160, 161, 174, 205, 238, 242–244, 308, 320, 335, 344, 353, 355, 363, 364, 368, 371, 415, 439
 Atomic substitutivity rule, 238, 239, 335, 344, 361
 Augmented frame, 174
 Avicenna, 291
 Axiom, 4, 12–14, 20–22, 26, 56, 96, 99, 100, 102, 105, 109, 110, 112, 113, 216, 220, 222, 228, 262
 Axiomatic completeness, 29, 31, 41, 104, 215
 Axiomatic consequence, 46
 Axiomatic derivation, 47, 90, 102
 Axiomatic proof, 1, 9, 12, 19, 22, 24, 27, 29, 31, 45, 96–98, 131, 217
 Axiomatic soundness, 1, 14, 31, 45, 48, 103, 223
 Axiomatic theorem, 96
 Axiom of Necessity, 73
 Axiom of Possibility, 55
 Axiom scheme, 12–16, 20, 21, 24, 25, 28, 29, 86, 95, 98–104, 108, 110, 117, 119, 216, 219, 221
 Axiom system, vii, 4, 12–16, 20, 24–31, 35, 38, 43, 46–49, 51, 84, 85, 95–119, 125, 131, 215–218, 221, 222

B

Baptism, 395
 Baptismal ceremony, 388, 395
 Barcan, viii, 187, 220, 268–270

- Barcan formula, viii, 182, 186–193, 197, 200, 201, 210, 218–221, 267–271, 273, 275, 276
 Bare particular, 234, 235
 Basic Necessity Rule, 123, 124, 128, 133, 147, 152, 154
Bedeutung, 230, 231
Begriffsschrift, 54, 73
 Being, 258, 263–265
 Belief, 52, 112–114, 119, 169, 260
 Berlin, I., 259
 Binary, 5, 10, 33, 40, 79, 106, 107, 147, 149, 151, 160, 184, 227, 394
 Blackburn, P., 109, 153, 303
 Blamey, S., 413
 Boethius, 68
 Boolean valuation, 10, 12, 14, 15, 24, 26, 28, 29, 31, 33, 34, 36–42, 45, 47, 48, 79, 80, 107
 Bound occurrence, 161, 333
 Bound variable, 216, 262, 293
 Branch, 32–37, 40–42, 44, 122–124, 126–134, 140–145, 147, 149–152, 154, 195, 196, 198, 199, 202–210, 237–239, 241, 242, 248–250, 255, 256, 271–274, 334–336, 338–341, 344, 346–348, 358–362, 373, 424, 426–430, 432, 433, 436, 437, 446
 Branch conditions, 133, 199, 429
 Branch extension rule, 33, 37, 121, 122, 132, 154, 202, 241, 362
 Brentano, 265
 Bundle theory, 234, 235
 Buridan, J., 301, 302

C
 CA, 331, 336–341, 343, 347, 348, 362
 Canonical model, 107–110
 Carnap, R., 231, 261, 407, 422
 Cartwright, R., 258, 262
 Causality, 73
 Chain limit, 18
 Chellas, B., 98, 104
 Children, 32, 33, 122, 208
 Chrysippus, 68
 Church, A., 116, 291, 297, 311
c-interpretation, 398
 Closed, 33–36, 38, 41, 122, 123, 168, 359, 424, 430, 432
 Closed assumption, 34
 Closed branch, 33, 34, 123
 Closed formula, 34, 162, 168, 196, 221, 271, 358, 373
 Closure, 34, 39, 44, 127–129, 135, 138, 146, 155, 184, 196, 201, 203, 222, 239, 240, 243, 255, 338, 343, 345, 363, 385, 432, 433, 438
 CN, 331, 341–342, 348, 362, 373, 381, 382, 423, 426–429, 431, 432, 436–438, 441, 446
 Compactness, 14, 46–48, 90, 92
 Compatible, 112, 113, 192, 283, 284, 399, 401
 Complete Induction, 6, 11, 39, 40, 46, 78, 107, 136, 137, 161, 205, 237, 242, 246, 308, 356, 369, 372, 398, 403, 441
 Completeness, vii, ix, x, 1, 12, 14, 15, 28–29, 31, 40–48, 90, 92, 103–110, 131, 134, 139–148, 203, 205, 207–215, 218, 222, 239, 248, 249, 351, 352, 361, 372–373, 375, 427, 428, 436, 438, 440, 446
 Conceivable, 58, 87, 262, 263, 416
 Conjunctive Rule, 33, 122, 124, 127–129, 131, 141, 143, 197, 200, 201, 240, 271–273, 343, 345, 346, 385
 Connectives, 4, 5, 9, 10, 12, 20, 24–29, 33, 71, 80, 81, 103, 111, 114, 122, 132, 149, 159, 160, 184, 205, 320, 355, 385, 424
 Consequence, 11, 13, 16–19, 29, 30, 32, 35, 46, 47, 84, 90, 92, 101, 103, 105, 106, 112, 123, 130, 150, 193, 244, 304, 317, 333, 339, 342, 364, 366, 387, 406, 419, 424, 428, 442
 Consistency, 15–19, 29–30, 38, 39, 43, 100, 105, 106, 110, 135, 145, 212–214
 Consistent, 15–19, 24–29, 38, 39, 43, 44, 48, 57, 70, 105–107, 109, 114, 144–147, 212–214, 263, 292, 372
 Constant domain, 172–176, 178–180, 183–186, 188, 190, 193, 197, 199, 201, 203, 206, 209, 210, 212, 215, 217–221, 236–240, 242, 248, 253, 268, 269, 272, 273, 276, 307, 309, 321, 331–333, 335–345, 347, 351, 378, 402, 419, 422, 423, 425, 433–436, 438, 440–442
 Constant domain existential rule, 196
 Constant domain frame, 190
 Constant domain modal tableaux, 196–199
 Constant domain model, viii, 173–179, 181, 185, 191–193, 196, 236, 253–255, 269, 309, 311, 319, 336, 425, 440
 Constant domain proof, 272, 342, 434, 435
 Constant domain tableaux, 197, 239, 273, 274, 276, 335–342, 345, 425, 434

- Constant domain universal rule, 196
 Constant symbol, 279, 281–286, 299, 307–314,
 318, 320–324, 326, 327, 332–334,
 336–338, 341–345, 347–348,
 352–355, 359, 363, 365, 368, 371,
 372, 377, 378, 381, 384, 385,
 397–400, 402, 404, 405, 407, 411,
 413, 419, 426, 434, 442, 443
 Context, viii, 9, 10, 92, 96, 99, 134, 153, 159,
 160, 167, 171, 196, 218, 220, 221,
 231, 236, 279, 287, 289, 294–296,
 300, 312, 329, 353–356, 358, 359,
 368, 380, 388, 396, 403, 407, 408,
 410, 422, 436, 437
 Contingent, 64, 65, 68, 100, 163–165, 167,
 168, 230, 233, 235, 288, 305, 395,
 396, 415
 Contingentists, 268
 Contradictories, 46, 56, 66, 116, 123, 183, 290,
 409, 415
 Contraries, 17, 54, 56, 66, 232, 235, 386
 Converse Barcan formula, viii, 186–193, 201,
 218–221, 269, 275, 276
 Counterpart, vii, 26, 92, 99, 118, 119, 123,
 130, 132, 171, 218, 279, 313, 343,
 383
 Crossley, J.N., 301, 302, 329
- D**
De dicto, 164–168, 170, 231, 232, 281,
 287–297, 299, 300, 302, 329, 394,
 399, 409, 411
 Deduction theorem, 19–25, 27, 30, 92, 93, 105,
 106
De facto rigid, 393
 Definite description, vii–xi, 51, 166, 167,
 231–233, 257, 259, 284–286, 296,
 307, 310, 328, 383, 385, 387, 394,
 395, 407–446
 Definite Description Rule, 427, 428, 430, 432,
 433, 436, 437, 446
 Deflationist, 259–263, 266, 267, 409
 Degree, 6, 39, 40, 78, 107, 136, 137, 161, 179,
 205, 290, 299, 308, 309, 364, 441,
 443, 445
De Interpretatione, 55, 64, 69
De jure rigid, 393
 Deontic logic, 86
 Deontic modality, 52
De re, 164–168, 170, 231, 232, 281, 287–296,
 299, 302, 304, 305, 329, 394, 399,
 411
 De Rijke, M., 109, 153, 303
- Derivation, 13, 14, 16–18, 20–24, 30, 35, 47,
 102, 106, 130, 131, 143, 148
 Derived Existence Rule, 427, 430
 Derived rule, 74, 96, 98, 239, 240, 256, 272,
 338–340, 342, 344, 427, 429, 431,
 440
 Derived Tableau Rule, 255, 256, 429, 433
 Descartes, 163
 Descendant, 208
 Description theory, 395
 Designates, viii, 110, 167, 171, 204, 230, 233,
 243, 254, 256, 266, 282–285, 287,
 293, 297, 298, 305, 307, 309–311,
 314–322, 324–328, 331–357,
 359–362, 365–368, 370, 373, 375,
 377–379, 381, 383–385, 387–391,
 393, 395, 397, 399, 401–405, 407,
 408, 410, 413–420, 422, 426, 428,
 435, 437, 442, 445, 446
 Designation, viii, ix, 167, 231, 282, 284–286,
 292, 298, 307, 320–321, 334–336,
 381, 383–391, 393, 403, 404,
 413–416, 419, 424, 426, 440
 Designation valid, 402–404
 Designator, 171, 292, 305, 394
 Determinism, 65
 Diodorean conditional, 71
 Diodorus, 63, 68–71
 Disjunctive Rule, 33, 122, 124, 127, 128, 131,
 143, 144, 197, 200, 273, 274
 Domain, viii, x, 165, 172–186, 188–193,
 195–215, 217–223, 229, 236–245,
 248–256, 264, 266–276, 281–283,
 295, 307, 309, 311, 315, 318, 319,
 321–328, 331–333, 335–356, 358,
 362, 365–371, 373, 375, 377, 378,
 381, 383–386, 388, 391, 396, 398,
 400–403, 413, 415–426, 433–438,
 440–445
 Domain of a frame, 180, 309, 315, 321, 396
 Domain of a model, 172, 174, 181, 183, 184,
 195, 206, 221, 229, 236, 244, 254,
 271, 328, 353, 368, 369, 371, 377
 Double negation rule, 111, 119, 433
 Doxastic modality, 52
 Dummett, M., 87
 Dynamic logic, 59, 404–406
- E**
E-complete, 144, 145, 212–214
 Einstein, 51
 Environment, 175
 Epictetus, 68

- Epistemic, 99, 100, 116, 289, 295, 380, 396
 Epistemic logic, 99, 112–116, 119, 192
 Epistemic modality, 52
 Equality, vii, viii, 171, 193, 225, 227–252, 254, 255, 269, 270, 275, 311, 335, 338, 340, 344, 347, 348, 351, 359, 361–363, 366, 367, 369, 370, 375, 377–382, 400, 427, 430, 432, 433, 439–441, 443, 444, 446
 Equality conditions, 244, 362, 363, 439
 Equality relation, 236, 241, 245, 254, 270, 377
 Equivalence class, 227, 228, 243, 245, 246, 251, 371, 372, 440, 443
 Equivalence relation, 85, 86, 99, 227–229, 243, 245, 371, 440, 441, 443
 Essential property, 163, 165, 168, 288, 396
 Euclidean, 85–88, 100, 104, 109, 110
 Evaluation, 294, 295, 327, 328, 353, 355, 356, 413, 414
 Evening star, viii, 230–233, 284, 379, 396
 Existence, viii, 16, 31, 44, 54, 79, 111, 163, 165, 184, 188, 192, 225, 237, 253–276, 291, 298, 309, 321, 335–337, 345, 346, 383–391, 396, 408, 409, 414, 415, 417, 419, 421, 424, 439, 440
 Existence abstract, 336
 Existence predicate, 184, 186, 257, 273, 327, 385, 386, 425, 426, 433
 Existence relativization, 184
 Existence Rule, 337–339, 341–343, 427, 440
 Existence supposition, 424
 Existential Rule, 196, 197, 199–201, 240, 255, 271–274, 346, 346, 423, 424, 432
- F**
 Fair algorithm, 41, 140, 207, 446
 False at a world, 81, 83, 162, 182, 415, 428
 Fine, K., 222, 294
 First-order formula, 160, 162, 295, 333
 First-order modal axiomatics, 215–223
 First-order modal formula, 159–162, 309, 411, 441
 First-order modal tableaux, 195–214
 First-order proof, 218
 Fishhook, 71
 Fitch, F., 116, 117
 Fitting, M., xi, 31, 80, 90, 98, 119, 121, 143, 172, 195, 222, 282, 308, 311, 316, 324, 325, 335, 430
 Formula, 5, 10, 77, 121, 160, 308, 313, 333, 411, 412
 Formula with definite descriptions, 411, 426
 Frame, 57, 60, 79, 84–91, 99, 104, 109, 125, 134, 163, 174, 179–181, 187, 188, 190–193, 219, 269, 270, 275, 309, 315, 321, 387, 396, 398
 Frame domain, 180
 Free logic, 172, 183–184, 255, 365, 366
 Free occurrence, 160–162, 196, 237, 238, 256, 313, 334, 356, 357, 360, 400, 429
 Free variable, 61, 62, 160–164, 167, 172, 175, 176, 180, 181, 183–185, 192, 195–197, 199, 204, 211, 212, 216, 218, 219, 221, 222, 225, 236, 240, 242, 254, 256, 269, 271, 294, 295, 308, 311, 313, 314, 319, 326, 333, 351–353, 356, 358, 369, 378, 386, 387, 396, 411, 412, 414, 426, 428, 429
 Frege, G., viii, 54, 55, 59, 73, 74, 230–233, 261, 262, 291, 297, 298, 395, 407, 413, 414, 422
 Function domain, 179, 245, 319, 371, 398, 403
 Function symbol, vii, viii, x, 159, 279, 281, 284, 285, 307, 312–323, 327, 331–334, 336–378, 384, 385, 397, 398, 401–403, 412, 413, 426, 439, 442, 443
 Future contingent, 65, 68
- G**
 Gabbay, D., 104
 General object term, 340, 345, 354
 General substitutivity rule, 238, 239, 361–362
 Generated, 5, 140, 191, 248, 334, 335, 338–342, 344, 347, 348, 359–361, 363, 364, 367–369, 371, 398, 426–430, 432, 438, 439, 446
 Global assumption, 91, 92, 101, 130, 143, 144, 193, 270–272, 275, 276, 431
 Global assumption rule, 130, 131, 148
 Global deduction, 92
 Global part, 102
 Gödel, 12, 96, 110–111, 308
- H**
 Heyting, A., 111
 Hintikka, J., xi, 38, 112–114, 118, 444
 Hintikka set, 38–40, 44–46, 134–139, 141, 142, 144, 147, 203–206, 209–211, 213, 214, 242–249, 362–371, 373, 438–442, 444–446

Hintikka's Lemma, 39, 122, 134, 136, 137,
204, 209, 214, 243, 247–249, 365,
367–370, 373, 440–446
Humberstone, I.L., 301, 302, 329
Hume, D., 73
Hybrid logic, 153–156, 303, 431

I

Identity, 51, 227, 230, 233, 234, 257, 266, 410
Indiscernibility of identicals, 229, 232–235
Induction, 5–7, 11, 22, 39, 40, 46, 78, 107,
108, 136–138, 161, 176, 185, 191,
205, 237, 242, 246, 247, 308,
356–358, 369, 370, 372, 398, 403,
441, 443–446
Infix notation, 236
Inflationist, 259, 263–267
Intensional term, 307, 313–315, 320, 411, 412
Interpretation, 174, 236, 245, 309, 315, 321,
368, 371, 398, 399, 413, 442, 443
Intuitionistic logic, 24, 110–111, 118
Iterated predicate abstracts, 309

J

Justification, 21, 74, 118–119, 239, 256, 428,
429, 432

K

Kamp, J.A.W., 294
Kant, I., 53, 54, 73, 230, 261
Kaplan, D., 105, 294, 295, 393
Knowability paradox, 116–117
Kripke, S., ix, vii, 58, 78, 84–86, 104, 105,
111, 174, 217, 218, 221–223, 233,
235, 388, 393–396, 424

L

Lambda calculus, 296, 297
Lambert, K., 183, 184
Large scope, 287
Law of bivalence, 68
Law of excluded middle, 65, 66, 68, 111, 408,
409
Law of noncontradiction, 232, 265
Leaves, 32, 139, 260, 430
Left child, 32, 33, 122
Leibniz, 53, 57
Leibniz's Law, 228
Lemmon, E., 105
Lemmon, J., 104

Lemmon Notes, 104, 105
Lesser modal cube, 125, 126, 130–134, 136,
139, 142, 145, 195, 202–204, 212,
239, 242, 243, 248, 335, 351
Lewis, C.I., 71, 72, 74, 86
Lewis, D., 171
Lincoln, A., 3, 380
Lindenbaum, 18, 104, 105
Lindenbaum's Lemma, 16–19, 28, 43, 104,
106
Linear, 88
Local assumption, 90–92, 101, 102, 106, 130,
131, 150
Local assumption rule, 130
Local deduction, 92
Local deduction theorem, 106
Locally constant domain, 191, 269
Locally rigid, 402–404
Locally strongly rigid, 397–401, 435
Local part, 102
Local rigidity, 397, 402, 403
Logical consequence, 35–36, 90, 101–103,
113, 130–131, 143–144, 192
Logical omniscience, 112, 113

M

Makinson, D., 105
Marcus, R.B., 74, 186, 187, 393
Master argument, 68–70
Material conditional, 71
Maximal consistency, 16, 17, 38, 108,
212–214, 248
Maximal consistent, 16, 144, 372, 446
Medieval interpretation, 68
Meinong, A., 264–266, 409
Meinong's schema, 265, 387
Mendelsohn, R.L., 230, 430
Metaphysical necessity, 388
Modal cube, 88–90, 98, 103, 107–109, 125,
131, 219, 227, 426
Modality, ix, 52–55, 59, 66, 68, 74, 84, 168,
169, 173, 186, 289–291, 394, 395,
433
Modal operator, 53, 56–59, 61, 62, 77, 78, 95,
112, 116, 118, 121, 131, 153, 154,
159, 166, 168, 186, 204–206, 231,
232, 283–285, 289, 292, 293, 296,
299, 303, 355, 380, 394, 404
Model, 78, 79, 107, 174, 180, 236, 310, 322,
354, 355, 322, 372
Modus ponens, 13, 20–22, 24, 26–29, 45, 70,
96, 97, 102, 103, 105, 109, 110, 112,
116–118, 216, 231

Modus tollens, 70
 Monotonic frame, 188, 219
 Monotonicity, 14, 47, 90, 91, 188–190, 218, 220, 253, 268–271, 275
 Monotonic model, 188, 269
 Morning star, viii, 230–233, 284, 379, 396
 Multi-modal, 59, 77, 78, 112

N
 Narrow scope, 287, 409
 Natural kind, 72, 396
 Necessitation, 73, 96, 101–103, 105, 106, 112, 116, 117, 219, 233, 269
 Necessitists, 268
 Necessity, ix, 55, 57, 65, 66, 71–75, 84, 99, 100, 105, 106, 140, 141, 164–170, 207, 222, 281, 285, 287, 290–292, 301, 302, 304, 377, 378, 388, 424
 Necessity rule, 123, 124, 126–129, 131, 133, 143, 147, 149, 152, 154, 197, 200, 201, 271–274, 341, 345, 424, 432, 433
 Negation rule, 122, 129
 Negative abstraction rule, 345, 385
 Negative abstract rule, 360–361
 Negative definite description rule, 428, 430, 432, 436, 437, 444
 Negative generation, 339, 426
 Negative introspection, 100
 Negatively generated, 334–335, 426–427
 Negative property, 286, 390, 391
 Negri style tableau, 151–153
 Newton, 51
 Node, 32–34, 41, 89, 109, 122, 140
 Nominals, 153–156
 Non-designation, ix, 285, 307, 321–327, 343, 347, 385, 414
 Nonexistence, viii, 256, 257, 272, 321, 385, 386, 389, 390, 415, 421
 Non-rigid, x, 282, 284, 299, 307, 309, 310, 312, 318, 319, 324, 327, 331–336, 342, 344, 352, 355, 363, 367, 371, 377, 378, 380, 394, 404–407, 413, 434, 442, 443
 Non-rigid designator, ix, 305, 379, 390, 414, 426
 Non-rigid function symbol, 307, 314, 317, 318, 322, 331, 352, 413, 442, 443
 Non-rigid model, 354, 355
 Normal modal logic, 73
 Normal model, 229, 236, 239, 241–246, 248, 249, 252, 269, 270, 367, 370, 371,

373, 377, 378, 384, 385, 400, 404, 436, 440, 444, 446
 Nozick, R., 115, 116
 Numerical identity, 227

O

Object existence, 346, 348
 Object term, 332–348, 351–369, 426, 438–441
 Ontological argument, 163, 260, 261, 308, 421
 Open assumption, 271
 Open branch, 35, 41, 130, 141, 206–209, 248–250, 373, 446
 Open formula, 164, 167, 169, 270, 271, 273, 386

P

Paradoxes, 71, 116–117, 258–260, 263, 264, 321, 409, 449
 Paradox of nonbeing, 257–262, 409
 Parameter, 161, 195–197, 199, 201–208, 210–216, 236, 238, 240, 242–246, 248, 249, 255, 256, 271, 272, 332–335, 338–340, 343–345, 348, 351–353, 358, 361–363, 365, 368, 369, 371, 426–430, 433, 437, 438, 440–445
 Parameter existence rule, 255, 271, 272
 Parameter nonexistence rule, 256, 272
 Parent, 32
 Parmenides, 257, 258
 Parmenides' Principle, 262–263
 Parsons, T., 265
 Partial function, 321, 322, 442
 Partiality, 320–322
 Partial model, 180
 Partition, 227, 443
 Path, 32, 191, 232, 257, 397, 403, 424
P-complete, 145–147, 213, 214
 Peripatetic school, 68
 Permutation, 166, 186, 192, 222
 Persistently rigid, 393
 Philo, 68, 71
 Philonian conditional, 71
 Plato, 257, 262
 Plato's Beard, 257
 Positive abstraction rule, 338, 345, 385
 Positive abstract rule, 359–361
 Positive definite description rule, 430, 432, 433, 437, 439, 443
 Positive generation, 438
 Positive introspection, 100

- Positively generated, 334–335, 339, 341, 342, 347, 348, 359–361, 363, 364, 367–369, 371, 426–430, 432, 438, 439, 446
- Positive property, 286, 321, 391
- Possibilist quantification, viii, 173, 186, 268, 413
- Possibilists, 173, 192, 253, 267, 268, 324
- Possibility rule, 123, 124, 127, 128, 131, 132, 141, 143, 145, 146, 149, 152, 154, 197, 200, 201, 240, 271–273, 338, 341, 345, 346, 424, 432, 433
- Possible state, 57
- Possible world, 57–58, 78–80
- Predicate, viii, x, 52, 71, 159, 168, 184–186, 228–230, 232, 254, 255, 257, 259–262, 273, 281–305, 307–352, 355, 356, 358, 359, 362, 365, 385–387, 389, 391, 394, 409, 410, 417, 419, 420, 425, 426, 433, 434
- Predicate abstract, 167, 292, 297–300, 308, 313–315, 320–322, 327, 334, 336, 337, 341, 365, 370, 385, 412, 416, 418, 442, 445
- Predicate abstraction, ix, viii, 166, 232, 266, 281–305, 307–350, 355, 356, 359–362, 364, 366, 367, 370, 372, 377–382, 385, 389, 390, 405, 427–430, 432, 433, 439, 441, 446
- Predicate abstraction rule, 334, 336, 339, 341, 343, 346, 347, 352, 359, 385, 427–430, 432, 433, 446
- Predicate symbol, 159, 254, 324, 420
- Prefix, 121, 122, 124, 130, 132, 133, 135–137, 140, 141, 146, 147, 153, 199, 201, 204, 206, 207, 210, 211, 213, 215, 237–239, 241, 243, 248, 255, 271, 273, 332–335, 338–340, 344, 345, 347, 348, 352, 353, 363, 364, 366–369, 373, 426, 443
- Prefix formula, 121, 123, 132–134, 136, 140–142, 145–147, 202, 204, 206–210, 212, 213, 242, 248, 249, 334, 352, 353, 358, 363, 367, 373, 437, 441, 443
- Prefix tableau, 121, 126, 134, 139, 147, 149, 153, 155, 195, 239, 334
- Prefix function, 132, 201, 202, 352–354, 356, 358, 359, 361, 362, 368, 436, 437
- Priest style tableau, 149–152
- Primary occurrence, 287, 297, 408
- Principle of plenitude, 70
- Prior, A.N., 52, 59, 63, 69, 70, 153, 192, 290, 298, 303
- Problematic, 52, 54, 66, 73, 106, 114, 168, 229, 257, 317, 379, 442
- Proof, 13, 34, 96, 123, 150, 152
- Proof procedure, ix, 3, 4, 31, 36, 114, 123, 125, 131
- Propositional formula, 5, 32, 36, 38, 41, 43, 78, 216
- Propositional logic, x, ix, vii, 4, 9, 11, 19, 20, 31, 38, 45–47, 51, 52, 77, 79, 80, 95, 105, 107, 137, 159, 160, 217, 241, 286, 331
- Propositional modal formula, 77, 78, 135, 145
- Propositional modal model, 79
- Propositional proof, 40, 270
- Provability, 24, 34, 110, 111, 119, 209, 348, 404, 423
- Pseudo-closed, 197, 202, 203, 212, 213, 333–337, 341–344, 346, 347, 352, 354–356, 358–369, 371, 426, 439
- Q**
- Qualitative identity, 227
- Quantification, ix, viii, 173, 179, 183, 186, 197, 205–210, 216, 253–255, 268, 292, 331–332, 335, 340, 368, 386, 388, 413
- Quantified modal logic, 74, 126, 159–193, 231, 307, 311, 331, 351, 370, 441
- Quantifier, 58, 60, 159, 160, 166, 171, 173, 179, 183, 184, 186, 192, 199, 196, 203
- Quantitative identity, 227
- Quine, W.V.O., 59, 74, 79, 165, 168–170, 221, 231, 257, 260–262, 264, 266–268, 288, 289, 297, 386, 411
- R**
- Realist, 72, 74
- Realization, 118, 119, 261
- Real world, 57, 58, 365, 416, 417
- Reductio, 111, 114, 260
- Reference, viii, 31, 61, 116, 127, 155, 177, 184, 230, 231, 259, 262, 263, 291, 309, 325, 388, 395, 414
- Reflexive, 63, 85–88, 99, 109, 111, 133, 138, 227, 228, 245, 301
- Reflexivity rule, 238–241, 335, 338–340, 343, 344, 347, 348, 361, 427
- Refutation system, 32
- Regularity, 96, 97, 101, 107, 148
- Relation symbol, 149, 159, 160, 174, 176, 180, 181, 184, 185, 204, 205, 210, 223,

- 236, 244, 245, 251, 283, 308, 309,
312, 314, 316, 319, 321, 354, 368,
369, 371, 384, 398, 399, 412, 415,
417, 420, 442, 443
- Relevant world, 397
- Replacement, 97–100, 239, 322, 335, 353,
402
- Right child, 32, 33, 122
- Rigid, ix, 282, 299, 312, 317–319, 324, 333,
352, 353, 388, 390, 393–404, 419
- Rigid designator, 282, 299, 305, 317, 324, 388,
390, 393–396
- Rigidifier, 305
- Rigidity, viii, 299, 317, 318, 388, 393–406
- Root node, 32, 41
- Routley, R., 265
- Rule of derivation, 20, 24, 27, 105
- Rule of inference, 69, 96, 105, 110, 216
- Russell, B., xi, 166, 231, 259–261, 263–266,
287, 289, 296–300, 385, 389, 394,
395, 407–414, 419, 422–426, 432
- Russell paraphrase, 422–426
- Russell's Schema, 265
- Ryle, G., 261
- S**
- Salmon, N., 393
- Satisfaction operator, 153
- Satisfiable, 36–39, 41, 44, 46, 131–134,
136, 139, 141, 142, 147, 202–204,
209, 214, 241–244, 248, 249, 318,
358–362, 365, 367, 368, 370, 371,
382, 417, 420, 436, 437, 440, 441
- Saturated, 38
- Schema, 24, 25, 56, 95, 99, 100, 117, 118, 220,
222, 228, 265, 268, 269, 275, 387,
399–401
- Scope, 60, 65–67, 160, 162, 165, 166, 231,
256, 279, 281, 285–287, 289,
292–300, 302–304, 313, 327, 329,
333, 377–379, 390, 394, 396,
408–411, 414
- Scott, D., 104, 105, 410, 419
- Searle, J., 262
- Secondary occurrence, 287, 297
- Segerberg, K., 55, 104
- Semantic consequence, 11, 12, 90–93
- Sense, 29, 40, 43, 47, 57, 63, 66, 71, 72,
92, 107, 113, 119, 126, 144, 148,
150, 161, 168, 180, 184, 186, 225,
229–231, 236, 244, 257, 261–263,
265–267, 282, 290, 291, 293, 298,
312, 313, 317, 324, 363, 371, 373,
378, 391, 397, 404, 405, 414, 430,
435
- Sentence, 56, 59, 60, 71–74, 114, 162, 164,
167–170, 175–179, 182, 183, 185,
191–193, 196–199, 201, 208, 209,
212, 214, 216, 222, 231, 239,
241, 248, 252, 256, 259–261, 269,
282–286, 289, 292–301, 321, 330,
338, 342, 379–382, 386, 387, 391,
394, 395, 405–408, 410, 413, 415,
417, 419–421, 425, 432, 435, 446
- Sequent calculi, 151, 153
- Serial, 85–88, 100, 104, 133
- Set model, 114, 115, 119
- Sheffer stroke, 28, 424
- Sinn*, 230, 231, 291
- Skeleton, 174, 179, 180, 188–191, 220, 254,
269, 270, 309, 310, 321, 325, 398,
399, 441
- Small scope, 410
- Smullyan, A.F., 232, 296, 411
- Smullyan, R.M., 31, 80, 172, 195
- Soames, S., 396
- Socrates, 257, 292
- Sophist*, 257, 262
- Soundness, vii, ix, x, 1, 12, 14, 31, 36, 37, 42,
45, 46, 48, 90, 92, 103–104, 109,
131, 134, 143, 147, 148, 197, 201,
223, 239, 241, 351, 352, 358–362,
375, 425, 435–438
- Special Necessity Rule, 126–128, 133
- Square of Opposition, 53, 55, 56, 61–65
- Stalnaker, R., 297, 308
- Stoic School, 68
- Strawson, P.F., 262
- Street, T., 291
- Strict implication, 72
- Strong completeness, 46–48
- Strongly rigid, 393, 397–401, 435
- Strong Recommendation, 425–426, 433, 436
- St. Thomas, 292
- Subalternative, 56, 64
- Subcontrary, 56
- Sublogic, 87, 100
- Subscript, 112, 201, 203, 204, 206, 207, 210,
236, 238, 244, 255, 332, 333, 352,
365, 368, 440, 441
- Substitutability, 162, 429
- Substitution, 67, 74, 95, 161, 238, 334, 410,
411
- Substitution Principle, 232
- Substitutivity Rule, 239–241, 335, 347
- Superscript, 333, 352
- Supervaluation, 68

- Symmetric, 63, 85–87, 100, 104, 109, 133, 134, 138, 139, 171, 220, 227, 229, 245, 275
- Synthetic proof procedure, 4
- Systematic construction, 206, 207, 248
- Systematic tableau construction, 105, 148, 206–209, 248, 249, 372, 373, 446
- T**
- Tableau, 32, 41, 121–123, 130, 140, 148, 151
- Tableau completeness, 38–46, 48, 105, 139–148, 205–210, 248–249, 372, 373
- Tableau consequence, 35, 46, 48
- Tableau proof, ix, xi, 31–35, 37, 38, 40, 42, 44, 45, 105, 114, 121–124, 127, 130, 134, 141, 144, 145, 149, 154, 155, 159, 195–197, 199, 201, 205, 206, 209, 214, 240, 248, 252, 255–256, 269–271, 274, 276, 332, 333, 338, 340, 342, 343, 345, 346, 348–351, 358, 359, 361, 365, 373, 380, 382, 384, 386, 401, 423, 431–435, 446
- Tableau rule, x, xi, 34–37, 39, 41, 97, 114, 123, 125, 126, 129–132, 134, 140, 149, 150, 152, 154, 195, 197, 201, 210, 237, 241, 248, 272, 327, 332, 334–350, 358, 359, 364, 381, 420, 426, 428, 436, 446
- Tableau soundness, 36–38, 131–134, 201–203, 241–242, 351–373, 436
- Tableau system, vii, ix, x, 1, 31, 32, 38, 46, 49, 84, 95, 104, 105, 121, 123–131, 139, 141–143, 147–149, 151, 152, 154, 155, 196, 201, 215, 238, 239, 241, 351, 366, 425, 428, 430, 431, 433, 434, 436, 437
- Tableau theorem, 35, 372
- Tarski, A., 20, 168, 175
- Tautology, 1, 11, 12, 14, 15, 25, 28, 29, 33, 36–38, 41, 42, 44–47, 51, 95, 97, 98, 103, 105, 112, 130, 216, 217, 222
- Temporal logic, 63, 153, 303
- Temporal modality, 433
- Tense operator, 61, 70
- Term equality, 269
- Term evaluation, 414
- Theory of descriptions, 395
- Thomason, R., 297, 308
- Tide table, 312, 318, 319, 327, 328
- Transitive, 63, 83, 85–88, 100, 104, 108, 109, 111, 133, 134, 138, 139, 171, 227, 229, 245
- Transworld identity, 233, 234
- Tree, 31, 32, 52, 122, 126, 208
- True at a world, 79–81, 83, 103, 122, 132, 142, 144, 153, 252, 254
- True in a model, 244, 382
- True in a non-rigid model, 354, 355
- Truth, 3, 9–12, 32, 35, 52, 56, 57, 59, 60, 65, 67, 68, 70, 72, 73, 79, 80, 84, 96, 100, 111, 114, 116, 117, 119, 153, 162, 164, 166–168, 170, 175, 176, 180, 181, 187, 190, 225, 232, 233, 247, 257, 264, 284, 286–288, 300, 303–305, 310, 315, 320, 322, 353, 386, 388, 391, 394, 395, 398, 403, 407, 409, 410, 413–415, 437, 450
- Truth conditions, 72
- Truth in a model, 80, 315, 322
- Truth Lemma, 107, 108
- Truth tables, 1, 9–12, 15, 31, 40, 47, 51, 59, 79, 80
- U**
- Unactualized possibles, 266–268
- Unary, 5, 10, 77, 78, 107, 319
- Universal distributivity, 216
- Universal generalization, 60, 216
- Universal Instantiation, 56, 171, 173, 216, 221, 222, 254, 298, 433
- Universal Rule, 196, 197, 199–201, 256, 271, 273, 340, 341, 345, 346, 424
- Use/mention, 71, 74
- V**
- VA, 331, 343–348, 362, 380, 401
- Vacuous quantification, 216, 335
- Valid, 3, 12, 84, 86, 87, 104
- Valid in a frame, 84, 163
- Valid in a model, 84, 91, 103, 176–178, 182, 193, 398, 401
- Valuation, 42, 47, 175–177, 180–182, 185, 188, 189, 191, 192, 201–204, 211, 218, 221, 241, 242, 244–247, 254, 283, 315, 319, 326, 352, 357–362, 368, 369, 371, 372, 398, 401–403, 416, 420, 423, 436, 437
- Variable, viii, 72, 107, 137, 151, 152, 159–161, 163, 175–177, 181, 191, 196, 203, 228, 232, 233, 245, 246, 254–256, 284, 294, 297, 298, 301, 304, 305, 308, 312–315, 318, 328, 329, 333, 353–357, 369, 378, 396, 400, 411, 412, 419, 422, 428

- Variant, 66, 100, 172, 175, 181, 354, 355, 369
 - Varying domain, 172, 173, 176, 179–181, 183–186, 188–193, 195, 199, 200, 203–204, 206, 209–211, 214, 215, 217, 218, 221–223, 236–243, 248, 249, 252–256, 264, 266, 268–274, 276, 307, 331–333, 335, 342–350, 358, 362, 365, 366, 370, 375, 381, 384–386, 400, 401, 413, 415–418, 420, 422, 423, 425, 426, 434–436, 438, 440
 - Varying domain existential rule, 199
 - Varying domain frame, 180
 - Varying domain model, viii, 173, 176, 179–186, 188, 191, 192, 201, 204, 209, 211, 212, 214, 215, 218, 221, 223, 236, 243, 254, 255, 268, 309, 311, 321, 326, 327, 335, 353, 356, 367, 383, 386, 415, 419, 421, 423–425
 - Varying domain proof, 222, 255, 434, 435
 - Varying domain tableau (VDT), 199–203, 209, 212–214, 239, 241, 255, 256, 270, 272, 273, 335, 342–350
 - Varying domain universal rule, 199
 - Venema, Y., 109, 153, 303
 - VN, 331, 347–353, 358–367, 370, 372–373, 377, 380, 382, 384–386, 402, 436, 438, 440
 - von Wright, G.H., 86, 290
- W**
- Watts, I., 52
 - Whitehead, A.N., 166, 287, 296–298, 408, 411, 412, 414, 424
 - Wide scope, 396, 409
 - Witness world, 365–367, 370, 371, 373
 - Wittgenstein, L., 230, 232, 262
- Z**
- Zeno, 68